DESIGN, IMPLEMENTATION AND VALIDATION OF A REAL-TIME DIGITAL SIMULATOR FOR PROTECTION RELAY TESTING

M. Kezunovic, Senior Member
J. Domaszewicz, Student Member
Texas A&M University

V. Skendzic[1], Member
Cooper Power Systems

M. Aganagic[1]
Siemens Empros

J. K. Bladow, Member, S. M. McKenna
D. M. Hamai
Western Area Power Administration

U.S.A.

Abstract - This paper presents implementation details describing a new design of a real-time digital simulator for relay testing. The simulator is developed under requirements to utilize as much as possible the low cost commercial computer hardware and system software support. The other requirements ask for a three-terminal support with extensive Graphical User Interface (GUI) and the simulation time step as low as 50μs. Such a design has been built for Western Area Power Administration (WAPA) and fully validated using a model of an actual WAPA system section. The simulator was delivered to WAPA in March 1994.

Keywords: Real-time Digital Simulator, Electromagnetic Transients Program (EMTP), Relay Testing

INTRODUCTION

The use of power system simulators for relay testing has been known for a long time. The well established approach is to use analog scaled power system models [1,2] or hybrid electronic simulators [3,4]. The latest trend is to use digital simulators that offer additional flexibility at a lower cost [5]. In the last ten years several digital open-loop simulators for relay testing have been constructed using low cost computers [6-11]. The main property of these designs was their open-loop mode of operation that could not directly support real-time interactions between the simulated power system and a relay under test.

The next generation of digital simulators for relay testing is aimed at real-time operation. It has been recognized that the real-time simulation of network response during faults is computationally involved and requires powerful computer facilities [12-15]. It has been proven that parallel computer architecture provides the performance required for real-time relay testing [16].

This paper reports on a new approach to real-time simulator implementation using a single processor computer for network simulation, and multiple Digital Signal Processors (DSPs) for instrument transformer and circuit breaker model implementation. The design has been optimized for power system protection device studies, and is capable of simultaneously supporting up to 3 independent test terminals. Simulator setup is shown in Figure 1.

The paper concentrates on the simulator implementation and validation details while the issues related to real-time network simulation and architecture concepts have been reported separately [17-20]. This design utilizes the waveform reconstruction and power amplifier subsystems developed under a separate simulator project funded by the Electric Power Research Institute(EPRI) [21-23]. A number of the results published as an outcome of the EPRI project are also utilized in the area of instrument transformer modeling [24-27].

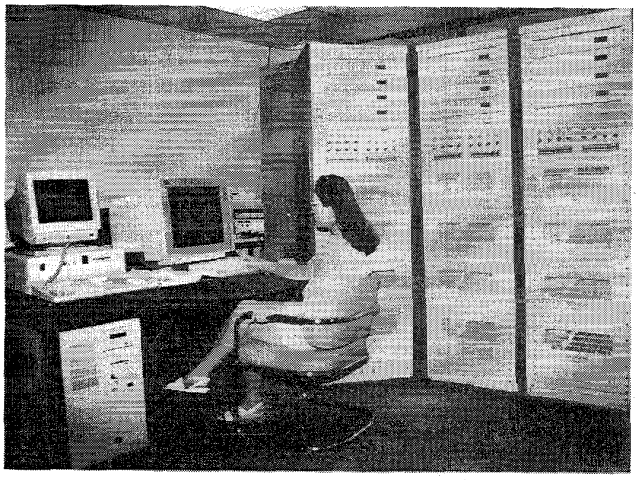

Figure 1. Real-time Power System Simulator Equipment

The paper starts with a discussion of the real-time simulator design requirements, presents the specific approach to real-time simulation and gives implementation details. Performance assessment and validation results are given at the end.

REAL-TIME SIMULATOR DESIGN REQUIREMENTS

Several design requirements were specified by the sponsor as well as by the research team. It was specified that the simulator must:

- Utilize low cost commercial hardware and system software to maximum extent possible
- Enable adequate system modeling and fault simulation complexity
- Provide sufficient output signal bandwidth
- Support real-time interaction between the simulated power system and the external devices under test
- Perform reliable monitoring of real-time program execution

Utilization of commercial low cost solution was extensively studied during Phase I of the project where feasibility of using commercially available parallel processing architecture was evaluated. At that time (1991) it was concluded that the readily available commercial parallel architectures were not suitable for the given application. The remaining choice was to utilize the existing high performance single processor computers for the simulator implementation. This design option has been specified as a requirement for the simulator implementation.

Adequate system modeling complexity has been translated into requirements for detailed modeling of transmission lines and nonlinear elements as well as precise representation of instrument

95 WM 034-9 PWRD A paper recommended and approved by the IEEE Power System Relaying Committee of the IEEE Power Engineering Society for presentation at the 1995 IEEE/PES Winter Meeting, January 29, to February 2, 1995, New York, NY. Manuscript submitted August 1, 1994; made available for printing November 30, 1994.

[1] The work was done while the authors were with Texas A&M University

transformers and circuit breaker responses. The fault simulation had to represent various fault events on transmission lines that can be represented with π models, constant parameter (cp) models and frequency dependent (fd) parameter models. Further more, the fault waveforms had to be generated for the lines containing series capacitors with Metal Oxide Varistors (MOVs) and surge arresters. All of the faults had to be generated with up to three relay controlled circuit breakers, enabling the testing of three independent relays interacting with a given network at the same time.

Output signal bandwidth is directly related to the simulation step size. As indicated in [28], precision of the power system simulation process depends on the excitation signal frequency. Error associated with the trapezoidal integration process increases as the signal frequency approaches Nyquist limit: $f=f_S/2=1/(2t_S)$. In order to keep the output error below ≈10%, the simulation sampling frequency should be at least 10 times higher than the desired output signal bandwidth. For fundamental frequency based protection equipment testing, the required bandwidth is typically >500 Hz [29], making it necessary to perform the simulations with f_S>5kHz (t_S<100μs). This limit applies as long as the trapezoidal integration rule is used. Others are possible.

Real-time interaction between the power system and the relays under test is well localized involving two distinct interface points. First is the circuit breaker used to perform on-line modifications of the power system network topology. The breakers are normally controlled by the externally connected relays under test. Second, is the Instrument Transformer (IT) secondary used to feed analog signals to the relays. A straight forward solution based on the closed-loop feedback design is very costly to implement. It can however be replaced by using a first order (linear) approximation of the relaying burden, added to the precise instrument transformer models [25]. This approach was selected for the RTS simulator implementation. Real-time interaction requirements are reduced to:

- Generation of secondary side curr. and volt. waveforms
- Acceptance of breaker commands (open/close)
- Generation of breaker auxiliary contact outputs

Reliable real-time program execution monitoring (Timing correctness verification) must be designed into the system. Following is a list of the main real-time power system simulation properties that were set as criteria for the simulator design:

- Failure to meet timing deadlines can be tolerated if the occurrence is followed by a prompt operator notification and an immediate system shutdown.
- Simulation task list is cyclic and can be executed with a simple repetition of a predefined task pattern.
- There is no need to request explicit time guarantees before attempting a given real-time operation.
- Timing constraints can be imposed and verified externally by an independent I/O system hardware.

The failure to successfully complete some of the runs is tolerable as long as the system guarantees that the condition is properly detected, and the system can be safely shut down.

SIMULATION PROBLEM PROPERTIES

Following is the list of main power system simulation problem properties used to streamline the real-time simulator operation:

- Transmission lines with modal domain propagation times longer than the simulation time step partition the simulated system into a number of smaller decoupled sub networks. This effect has long been recognized [28], but has not been extensively used in the original EMTP design. The conductance matrix is block diagonal as shown in Figure 2.

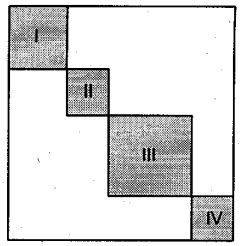

Figure 2. Conductance Matrix Structure

Dynamic changes in network topology as a result of circuit breaker operations, call for updating of the admittance matrix in real time. The most severe consequence of this fact is that the amount of computation as a function of the system size n (number of nodes), increases at a a rate that is higher than that of any linear functional. This clearly provides an indication as to the possible benefits of exploiting the natural decoupling of the system across the transmission lines.

As indicated in the literature [12-16] each of the sub-networks can be assigned to a separate processor, providing natural code parallelization. Similar savings are applicable to single processor implementations [17,18].

- In general, the presence of nonlinear elements in a network necessitates application of an iterative process at each simulation time step. The process involves repeated solution of the linearized nodal equations within a single time step. Moreover, at each iteration, the underlying matrix, as well as the right hand side in the conventional EMTP approach are updated. The Real Time System (RTS) requirements for speed of computation are such that an alternative approach had to be developed [17]. The main characteristics of the approach are:

  a. Focus on a specific type of nonlinear components: Metal Oxide Varistors, highly nonlinear elements used for protection of series compensation capacitors [19].
  b. Modeling the nonlinear element and the surrounding circuitry as a single complex component.
  c. Voltage, instead of the matrix updating scheme was developed. Consequently, only a linear (rather than quadratic) increase in the number of arithmetic operations is observed (with respect to the size of the corresponding block of the conductance matrix).

- Instrument Transformer (IT) models can be decoupled from the primary power system network by neglecting or approximating their mutual influence. Simulations were run in order to obtain the CCVT primary side impedance as a function of frequency and the load applied to its secondary. Results showing the behavior of a typical CCVT device are given in Figure 3.

The graph contains 4 curves representing the simulation results obtained with the following load impedance:

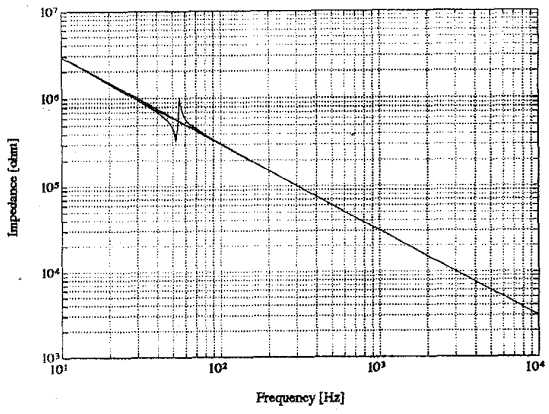

Figure 3. Input Impedance of a Typical CCVT as a Function of Frequency and Load Level

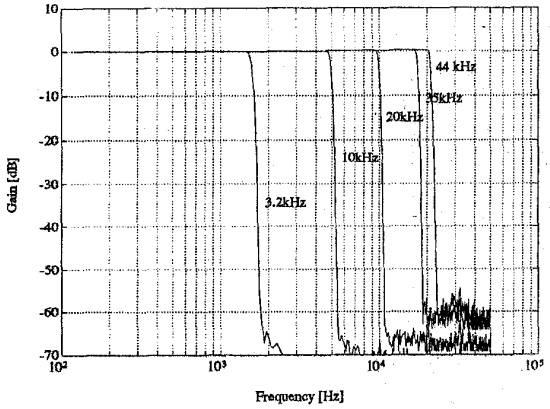

Figure 4. Waveform Reconstruction System Frequency Response Magnitude as a Function of Sampling Frequency

- shorted secondary (R_b = 1e-6Ω)
- R_b = 10Ω
- R_b = 100Ω
- open secondary (R_b = 1e6Ω)

All simulations were performed on the 115 V output tap. The results show almost ideal capacitive behavior. The main deviation is observed at the primary circuit resonant frequency (60Hz), and is most severe in the case of shorted secondary terminals. In all other cases, starting with an unrealistically low 10Ω load impedance, the deviation from the ideal (straight line) is negligible, and becomes lower as the load impedance increases. This makes it possible to decouple the IT models from the primary network, and to execute their simulation on a separate processor. In the case of CCVTs, input impedance can either be neglected or approximated by adding a capacitor to the primary simulation network. The capacitor value should be equal to the simulated device stack capacitance. Similar discussion applies to the Potential Transformer (PT), and Current Transformer (CT) models.

- Physical limitations associated with the mechanical circuit breaker design result in significant time delays between the moment the device receives a command and the moment it actually starts moving its mechanical components. The delays vary from 10 to 50ms depending on the breaker design. These delays can be efficiently used to optimize the real-time system operation.

- The time needed to calculate the system solution for a single time step will vary in proportion with the simulated system complexity. To achieve real-time operation, the simulation time step must be adjusted to match the actual execution time. Although the time step could be fixed, it is more appropriate to enable the user to select the desired time step.

  In order to support this feature it was necessary to design a variable sampling frequency waveform reconstruction system capable of correctly interpolating the output waveforms across a wide range of sampling frequencies [30]. Variable sampling frequency support makes it possible to optimize the system operation for a given simulation run. Furthermore, it provides a transparent upgrade path capable of accommodating any performance increase obtained by future improvements of the simulation computer hardware.

The waveform reconstruction system uses an 8 times over sampling digital filter in combination with a brick-wall analog low-pass filter. It is capable of supporting an arbitrary sampling frequency in the 3.2 to 44 kHz. range. The out of band output signal attenuation exceeds 85dB at all frequencies over $f_s/2$. Measurement results showing the system frequency response magnitude as a function of sampling frequency are shown in Figure 4.

The noise floor increase visible at higher frequencies is caused by the limited resolution (12 bits) of the FFT analyzer used for the measurements. Measurements were performed at the voltage amplifier output terminals.

IMPLEMENTATION APPROACH

Real-time Program Model

Based on the outlined simulation problem properties it is possible to build a matching real-time program model capable of optimizing overall system performance [30]. The model is based on the following assumptions:

- The simulation consists of multiple tasks with different computation complexity (execution times).
- There are no assumptions about temporal relation between tasks; tasks are fully independent and may be executed on independent processors.
- Tasks communicate by exchanging messages; each task can communicate with an arbitrary number of peers.

Every task is characterized with its:

- Execution Time
- Input Data Set
- Latency

Synchronization among tasks is performed in a distributed fashion regardless of the actual task *execution time*. Each task is responsible for verifying the completeness of its *input data set* and will wait until it receives all of the data required for calculating the next simulation step. Once this is accomplished, the task will start executing, producing an appropriate output data set. Before repeating the operation the task is responsible for communicating the result to all of its peers. The returned results automatically represent an acknowledgment of the message frame received from a

given task. *Task latency* is defined as a number of output data frames that can be generated based on the current input data set. It can be associated with transmission lines, breaker models, or any other component containing inherent time delays.

In order to use those delays to optimize real-time system performance it is necessary to take them out of the tasks and allocate them for system synchronization purposes. A simple queuing model illustrating this approach is given in Figure 5.

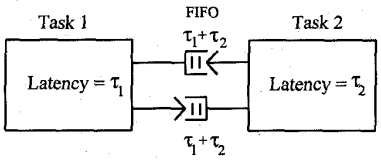

Figure 5. Simple Real-Time Program Queuing Model

It can be shown [30] that the system built according to this model guarantees distributed task synchronization. The maximum throughput will be determined by the slowest task in the network, and there will be no synchronization bottlenecks regardless of the actual network topology. To avoid the possibility of data loss, the intermediate First In First Out (FIFO) buffers must be dimensioned to accept a number of data frames equal to the addition of individual task latencies on the given data link.

Real-time Simulator Design Implementation

Based on the general queuing model it is possible to develop a particular real-time simulator implementation by taking into account various hardware constraints and application imposed requirements. An example showing the implemented real-time simulator program structure is given in Figure 6.

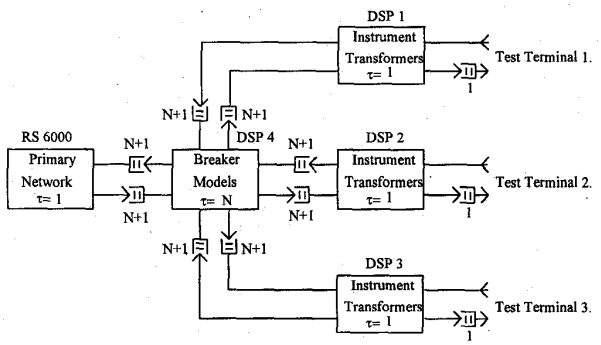

Figure 6. Real-Time Simulator Program Model

The design attempts to optimize the operation of a conventional high performance single processor RISC workstation, aided by 4 DSP processors with predefined task allocation.

The resulting hardware architecture is shown in Figure 7. It is easy to notice the close relationship between the actual power system devices and the functions allocated to various processors.

As can be seen from the figure, the system is physically partitioned into 3 protection terminals. Each terminal is equipped with its own hardware needed to scale down the primary side power system waveforms (IT models executing on 3 dedicated DSP processors), reconstruct the output signals (I/O subsystem), amplify them (amplifier subsystem), and acquire the relay response data (I/O subsystem). The fourth DSP processor and the IBM RS 6000/580 perform the primary network simulation, with a loosely coupled GUI workstation used to support user interaction.

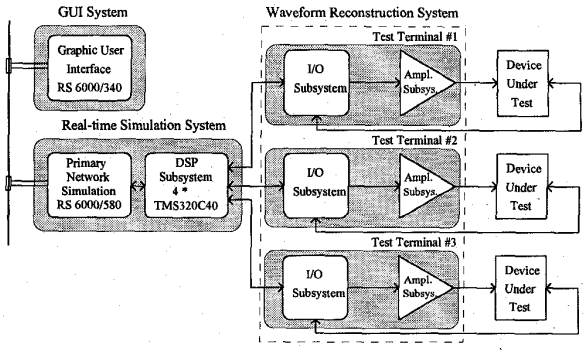

Figure 7. Real-Time Simulator Hardware Architecture

The real-time simulator timing structure is shown in Figure 8. Concentric circles shown in the figure indicate the timing constraint inheritance mechanism. Primary timing is derived from a single crystal controlled oscillator. It is then passed successively from outer towards inner layers. The outer circle devices are therefore responsible for verifying the timing correctness of their internal circle neighbors.

This structure exhibits an interesting property of allowing increase of task granularity towards the internal circles.

First Layer (I) supports the most stringent timing requirement of sending the simulator data to the D/A converters (every 50-100μs depending on the requested time step). To minimize the unwanted analog signal degradation, the actual time instant at which this operation is executed must be kept within ±1ns of the ideal sampling clock transition. Due to the stringent sampling clock jitter requirements, this layer requires hardware implementation. The simulator uses PLL clocks generated locally in each of the 3 output terminals, aided by 3 bit elastic buffers accommodating the time jitter present on the serial data links connecting the I/O terminals to the computer subsystem.

Error detection and failure recovery mechanisms are implemented by performing continuous serial link data format compliance monitoring. The hard-wired logic protects the rest of the system in case of serial link or DSP subsystem failure.

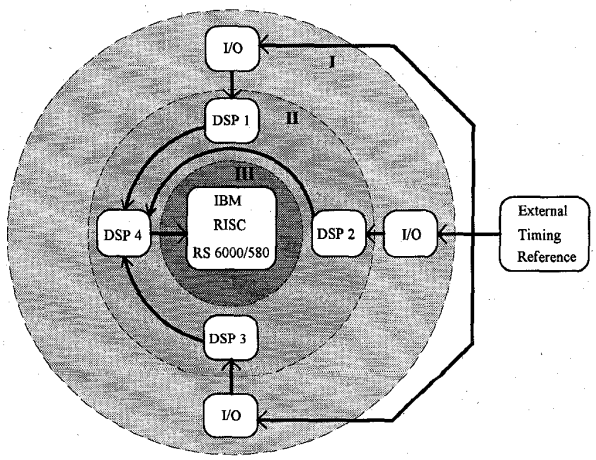

Figure 8. Real-Time Simulator Timing Structure

Second Layer (II) Contains 4 DSP processors. The timing requirements on this layer are relaxed by discarding the previously mentioned ±1ns jitter specification and replacing it by a 50-100μs time step requirement. The sampling clock is still used as a main synchronization source. It is being transferred from the I/O systems to 3 DSP processors performing the instrument transformer modeling. Timely operation and mutual synchronization of these devices is crucial for correct system operation. It is performed by the fourth DSP processor which is in charge of breaker modeling as well as data transfer and distribution between the remaining 3 DSP processors and the RISC System/6000.

Third Layer (III) consists of the IBM RISC System/6000 executing the real time RTS simulation core. By using the breaker operating time delays it is possible to further relax the simulation timing requirements going up to a 10-50 ms level.

Operating System Support

As can be seen from the previous discussion, the use of breaker model delays effectively increases the RS6000 task granularity from 50μs to ≈5-30ms depending on the actual breaker parameters and the simulation time step. This increase brings the application within the reach of AIX operating system capabilities making it possible to use mainstream real-time operating system support [31]. However, the high computational demand imposed by the primary network simulation makes it necessary to suspend all concurrently running applications that may interrupt the real-time simulation run. This also applies to all background system services (communication system demons, mail, X, etc.). Page faults associated with the virtual memory support are eliminated by using AIX functions for loading and *pinning* real-time application in the physical memory space.

In the case of RTS, the design was taken one step further by removing the unnecessary operating system overhead. During the real-time execution, the AIX operating system is effectively suspended by disabling system interrupts. Since a typical power system simulation lasts only several seconds, the lack of system response perceived by other users during the simulation run was found to be acceptable.

DESIGN VERIFICATION

This section outlines the methodology used to measure, optimize, and validate the simulator real-time operation. The best way of starting the analysis is to look at the time needed for solving the primary power system network model.

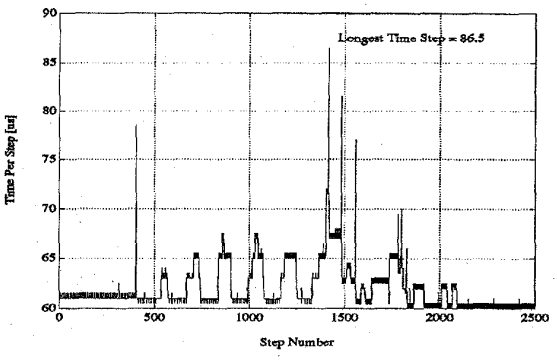

Figure 9. RTS Time Step Calculation Loop Profiling Results

Typical results showing the actual measurements obtained on the RS6000/580 to TMS320C40 communication link are shown in Figure 9. Each point on the graph represents the amount of time needed to calculate a single step solution and transfer the results to the DSP subsystem. As already expected, the computation time varies significantly throughout the simulation run. By comparing Figure 9. with the simulator output waveforms, and the resulting event log, it was possible to determine that for a given network configuration, every breaker (and/or time controlled switch) operation introduces a one step time penalty in the order of ≈18μs. MOV operation is somewhat cheaper ≈3μs per device, but extends over the entire period of MOV activity, with additional possibility of having multiple MOV devices operating simultaneously.

Using these times in a simple real-time design would for a given case result with the shortest simulation time step being higher or equal to 87μs. The limitation can be alleviated by using inherent circuit breaker delays to optimize the system real-time operation. The performance gain will depend on the actual breaker model implementation. RTS currently supports up to 9 relay controlled breaker poles (3 breaker poles allocated to each protection test terminal), with a block diagram given in Figure 10.

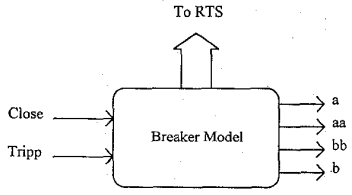

Figure 10. Circuit Breaker Block Diagram

All breakers are simulated as independent single phase devices with two inputs (Open, Close), and 4 auxiliary contact outputs (a, aa, b, bb). Timing diagram illustrating RTS breaker behavior is shown in Figure 11. Times T1-4 represent the breaker operating times. Label *'To RTS'* is used to denote capability of independently controlling the instant at which the primary network simulation receives the command to actually modify the switch state.

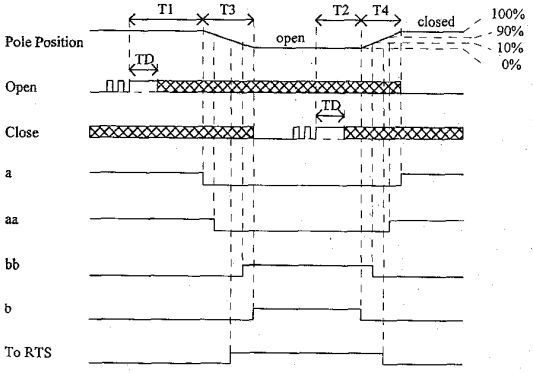

Figure 11. Typical RTS Breaker Operating Sequence

Depending on the type of the breaker, the delay can typically range from 1 to >5 cycles, giving an ample time margin for additional computation load balancing.

The benefits can best be demonstrated by using an example. Let us assume that a user wants to simulate a 2 cycle breaker with 'realistic' operating times based on the measurements reported in [32]. The sample breaker description file used for this example is given in Figure 12.

By using this model, the RTS operation will effectively be averaged over breaker mechanical response time (19.6ms) interval (>300 time steps). This means that for a given breaker model, the primary network simulation performed on the RS 6000 machine, can on average be run up to 300 samples ahead of the real-time (as seen on the simulator output). In the case of data shown in Figure 9, this means that the real-time simulation can successfully be executed with a 64.7μs time step. This was verified by running an actual real-time simulation with a 65μs time step.

17.8	17.8	17.8	Opening Delay Time	[ms]
9	9	9	Opening Travel Time	[ms]
18	18	18	Closing Delay Time	[ms]
15	15	15	Closing Travel Time	[ms]
20	20	20	Aux Contact 'a' (closed)	[% of Travel]
30	30	30	Aux Contact 'aa'	[% of Travel]
70	70	70	Aux Contact 'bb'	[% of Travel]
80	80	80	Aux Contact 'b' (open)	[% of Travel]
50	50	50	RTS Contact on Opening	[% of Travel]
50	50	50	RTS Contact on Closing	[% of Travel]

Figure 12. Breaker Description File Used in the Example

Results are very close to the long term execution times average. As already explained, the RTS can operate with an arbitrary output sampling frequency defined anywhere in the 3.2 to 44kHz range. The main question raised in such an environment is what is the maximum speed that can be achieved with a given simulation data set. Unfortunately, this information is normally not available beforehand.

One possible solution is to use a trial and error approach, with multiple simulation attempts targeted at finding a feasible set of operating conditions. Instead of relying on the operators experience on finding the shortest time step, it is much more convenient to make use of the fact that the demonstrated RTS architecture achieves exceptionally high RISC CPU utilization (over 90%), sustained throughout the entire real-time simulation run. As a consequence, it is convenient to generate two independent simulation program versions, one compiled for full real-time support, and other simplified by stripping off the I/O interface interaction and adding the standard time measurement services. The actual simulation time step is then estimated by simply running the non-real-time program version and using its mean execution time as a starting point for a consecutive real-time run. Measured time must be increased by ≈4-7μs to take into account the additional RS 6000 I/O transfer overhead. This utility is included as part of the standard RTS simulator software support.

CONCLUSIONS

The most interesting conclusions are:

- This design has demonstrated that a digital real-time simulator for relay testing can be built using the low cost commercial computer hardware and system software support.
- A simulation time step between 50-100μs is achievable for a quite complex network simulation including series capacitors with MOV protection and detailed models of instrument transformers.
- The simulator is designed having in mind future computer hardware upgrades. Selection of the wide spread commercial architecture makes it possible to piggy-back on the market driven technology developments. This makes the design a quite economic solution.

Acknowledgments

This development has been funded by DOE-Western Area Power Administration under contract FC65-90WA 07990. The authors want to acknowledge the participation of Drs. H. W. Dommel and J. Marti from the University of British Columbia, and Dr. A. Abur from the Texas A&M University in the Phase I of the project, involving the simulator feasibility study.

REFERENCES

[1] P. Muller, "Network Model as a Testing Device for Protection Systems Able to Simulate Various Types of Operating and Fault Conditions", *1980 CIGRE Session,* Paper No. 34-02, Paris, France, August 1980.

[2] R. E. Ray, H. J. Li. "A Computer-Directed Model Power System", *Western Protective Relaying Conference*, Spokane, Washington, October 1986.

[3] R. Joetten, et. al., "A New Real-Time Simulator For Power System Studies", *IEEE Trans. on Power Apparatus and Systems*, Vol. PAS-104, No. 9, pp. 2604-2611, 1985.

[4] G. Nimmersjo, et. al., "A Digitaly-Controlled, Real-Time, Analog Power-System Simulator for Closed-Loop Protective Relaying Testing", *IEEE Trans. on Power Delivery*, Vol. 3, No. 1, pp. 138-152, 1988.

[5] G. E. Alexander, et. al., "Analog vs. Digital Modeling of Power Systems", *Western Protective Relaying Conference*, Spokane, Washington, October 1989.

[6] A. Williams, R. H. J. Warren, "Method of Using Data From Computer Simulations to Test Protection Equipment", *IEE Proceedings*, Vol. 131, Pt. C, No. 7, pp. 349-356, 1984.

[7] J.Esztergalyos, et. al., "Digital Model Power System", *IEEE Computer Applications in Power*, Vol. 3, pp. 19-24, 1990.

[8] P. Bornard, et. al., "MORGAT: A Data Processing Program For Testing Transmission Line Protective Relays", *IEEE Trans. on Power Delivery*, Vol. 3, No. 4, pp. 1419-1426, 1988.

[9] M. A. Redfern, et. al., "A Personal Computer Based System for the Laboratory Evaluation of High Performance Power System Protection Relays, *IEEE Trans. on Power Delivery*, Vol.6, No.4, pp. 1402-1408, 1991.

[10] M. Koulischer, et. al., "A Comprehensive Hardware and Software Environment for the Development of Digital Protection Relays", *Proc. International Conference on Power System Protection*, Singapore, pp. 20-42, 1989.

[11] M. Kezunovic, et. al., "Dyna-Test Simulator for Relay Testing - Part II: Performance Evaluation", *IEEE Trans. on Power Delivery*, Vol. 7, No. 3, pp. 1097-1103, 1992.

[12] R. C. Durie, C. Pottle, "An Extensible Real-Time Digital Transient Network Analyzer", *IEEE Trans. on Power Systems*, Vol. 8, No. 1, pp. 84-89, 1993.

[13] D. M. Falcao, et. al., "Application of Parallel Processing Techniques to the Simulation of Power System Electromagnetic Transients", *IEEE Trans. on Power Systems*, Vol. 8, No. 1, pp. 90-96, 1993.

[14] K. Werlen, H. Glawitsch., "Computation of Transients by Parallel Processing", *IEEE Trans. on Power Delivery,* Vol. 8, No. 3, pp. 1579-1585, 1993.

[15] J. R. Marti, L. R. Linares, "Real-Time EMTP-Based Transient Simulation", *IEEE PES Summer Meeting*, Paper No. 93SM 511-6-PWRS, Vancouver, B. C., July 1993.

[16] P. G. McLaren, et. al., "A Real Time Digital Simulator for Testing Relays", *IEEE Trans. on Power Delivery*, Vol. 7, No. 1, pp. 207-213, 1992.

[17] M. Kezunovic, et. al., "Transients Computation for Relay Testing in Real-Time", *IEEE PES Summer Meeting*, Paper No. 93SM 383-0-PWRD, Vancouver, B. C., Canada, July 1993.

[18] M. Kezunovic, et. al., "A New Multiprocessor Architecture Implementation of a Real-time Simulator for Power System Applications", *ISCA International Conference on Computer Applications in Industry and Engineering*, Honolulu, Hawaii, December 1993.

[19] M. Kezunovic, et. al., "Computing Series Capacitors With MOV Protection in Real-time", *IEEE PES Summer Meeting*, Paper No. 94SM 400-02 PWRD, San Francisco, CA, 1994.

[20] M. Kezunovic, S. M. McKenna, "Real-Time Digital Simulator for Protective Relay Testing", *IEEE Computer Applications in Power*, Vol. 7, No. 3, 1994.

[21] M. Kezunovic, et. al., "Dyna-Test Simulator for Relay Testing - Part I: Design Characteristics", *IEEE Trans. on Power Delivery*, Vol. 6, No. 4, pp. 1423-1429, 1991.

[22] L. L. Mankoff, M. Kezunovic, "Protective Relay Workstation: Applications of a Digital Simulator", *Texas A & M Relay Conference*, College Station, TX, April 1991.

[23] M. Kezunovic, "Protective Relay Workstation - Application of Digital Simulator for Protective Relay Studies", *Electric Power Research Institute RP 3192-1, Phase I Final Report # TR-102781*, Palo Alto, CA, 1992.

[24] M. Kezunovic, et. al., "Digital Models of Coupling Capacitor Voltage Transformers for Protective Relay Transient Studies," *IEEE Trans. on Power Delivery*, Vol.7, No 4, pp. 1927-1935, 1992.

[25] Lj. Kojovic, et. al., "Computer Simulation of a Ferroresonance Suppression Circuit for Digital Modeling of Coupling Capacitor Voltage Transformers", *ISMM Conference on Computer Applications in Design Simulation and Analysis*, Orlando, Florida, March 1992.

[26] Lj. Kojovic, et. al., "A New Method for the CCVT Performance Analysis Using Field Measurements, Signal Processing and EMTP Modeling", *IEEE PES Winter Meeting*, Paper No. 94WM 003-4-PWRD, New York, January/February 1994.

[27] M. Kezunovic, et. al., "Experimental evaluation of EMTP-Based Current Transformer Models for Protective Relay Transient Study", *IEEE Trans. on Power Delivery*, Vol. 9, No.1, pp. 405-413, 1994.

[28] EPRI, "Electromagnetic Transients Program (EMTP)", Version 2. Revised Rule Book, Vol. 1:Main Program, EPRI EL-4541-CCMP, Palo Alto, California, March 1989.

[29] M. Kezunovic, et. al., "Digital Protection Relaying Algorithm Sensitivity Study and Evaluation", *IEEE Trans. on Power Delivery*, Vol. 3, No. 3, pp. 912-922, 1988.

[30] V. Skendzic, "Design Issues in a New Implementation Approach to Digital Fault Simulators for Protective Relay Studies", *Ph.D. Dissertation*, Texas A&M University, College Station, TX, 1994.

[31] *AIX Version 3.1 RISC System/6000 as a Real-Time System*, Publication # GC 24-3633-00, IBM International Technical Support Centers, Poughkeepsie, NY, 1992.

[32] R. O. Berglund, et. al., "One-Cycle Fault Interruption at 500 kV: System Benefits and Breaker Design", *IEEE Trans. on Power Apparatus and Systems*, Vol. PAS-93, pp. 1240-1251, 1974.

M. Kezunovic (S'77, M'80, SM'85) earned his Dipl. Ing. degree from the University of Sarajevo, Yugoslavia, the M.S. and Ph.D. degrees from the University of Kansas, all in electrical engineering in 1974, 1977, and 1980, respectively. Dr. Kezunovic's industrial experience is with Westinghouse Electric Corporation in the U.S.A., and the Energoinvest Company in Sarajevo. He also worked at the University of Sarajevo. He was a Visiting Associate Professor at Washington State University for the 1986-1987 Academic year. He has been with Texas A&M University since 1987 where he is an Associate Professor. He is a Registered Professional Engineer in the State of Texas.

J. Domaszewicz (S'89) earned his M.Sc.E.E. degree from Warsaw Technical University, Poland, in 1986. From 1986 to 1989 he was with the Department of Electronics at Warsaw Technical University. Mr. Domaszewicz is currently a graduate student at Texas A&M University.

V. Skendzic (M'88, S'90, M'94) holds a Dipl. Ing. degree from the University of Split, and M.S. degree from the University of Zagreb, Croatia, both in electrical engineering. He earned his Ph.D. degree from Texas A&M University, in 1994. From 1984 to 1989 he was a faculty member at the University of Split, Croatia. Dr. Skendzic is currently with Cooper Power Systems, Thomas A. Edison Technical Center.

M. Aganagic holds a Dipl. Ing. degree in electrical engineering, and M.S. degree in computer science. In 1978 he earned his Ph.D. degree in operations research from Stanford University. Until joining Texas A&M University as a research scientist in 1992, Dr. Aganagic was with the Energoinvest Company, Sarajevo, where he held different positions in the Research and Development Division. Since October 1993 Dr. Aganagic has been with Advanced Applications R&D Group of Siemens Empros. His main research interests lie in the areas of mathematical programming and its applications to power systems.

J. K. Bladow was born in Moorhead, Minnesota in 1959. He earned both Bachelor and Master degrees (power option) in electrical engineering from North Dakota State University. Mr. Bladow joined the Western Area Power Administration (Western) in January 1983, and has held various technical and managerial positions. Mr. Bladow currently is Western's Assistant Administrator for Washington Liaison. Mr. Bladow has been involved with Electromagnetic Transient Program (EMTP) studies, the application and setting of protective relays, system short-circuit studies, substation electrical design, and major electrical equipment specifications. Mr. Bladow is a member of the IEEE and a Registered Professional Engineer in Colorado.

S. M. McKenna was born in Spokane, Washington in 1951. He earned his B. S. degree from the Colorado School of Mines in 1978. Mr. McKenna joined the Western Area Power Administration (Western) in September 1980. Mr. McKenna is currently in the Substation Control Branch where he is involved in control system design for advanced technology devices, protective relaying application and testing, and power system testing. Mr. McKenna is a Registered Professional Engineer in Colorado.

D. M. Hamai was born in San Rafael, California on October 14, 1963. He earned his B.S. degree in electrical engineering from the University of Colorado in 1986. Mr. Hamai has been employed with the Western Area Power Administration since 1986 and is currently in the Technical Branch where he is involved with various Electromagnetic Transients Program (EMTP) studies, power system protection applications, and major power equipment requirements.