Computer Assisted Railway Control System for Sri Lanka Railways

N. Sayooran, J.S. Fernando, M.A. Mohamed, L.B. Thambugala and C.R. De Silva
Department of Computer Science and Engineering, University of Moratuwa, Sri Lanka.

***Abstract*— This paper describes about the design and implementation of the Computer Assisted Railway Control System for Sri Lanka Railways. Currently, the train traffic control system of Sri Lanka Railways has two control units. They are Outstation and Metropolitan control units. In the outstation unit, controllers spend more time in manually plotting the TDG (Time Distance Graph) than decision making and for reporting purposes the same data are re-entered. In the metropolitan unit, switch boards in illumination based panel is used to monitor and control train traffic. This panel has many drawbacks such as congested layout, unclear information and unreliable bulbs and switches. So at the end of the day, all these problems contribute to trains getting delayed. The proposed system is built using 4-tier networked architecture which addresses the aforementioned drawbacks and bottlenecks. Features such as TDG, TDG Designer, Track Designer, Dispatcher Control View, Rules Editor, Train Timetable, Traffic Control Playback, Auto Train Supervision and Reporting have helped to improve the overall operation of Centralized Train Traffic Controlling of Sri Lanka Railways. Moreover, the solution comprises a high performance and more reliable hardware unit with higher I/O capacity that provides a hardware interface between the proposed system and the existing electromechanical interlocking system. The solution was successfully tested at Maradana Control Office and Dematagoda Workshop with the presence of senior officials from Sri Lanka Railways. The test results obtained depict that the system has helped to improve the overall train controlling operations of Sri Lanka Railways and proved its potential for implementation.**

***Index Terms* - centralized control, control systems, rail traffic control, SCADA systems**

I. Introduction

Since the inception of Sri Lanka railways, the signaling and traffic controlling of trains are done in a centralized manner i.e. train dispatchers monitor and control train traffic from a centralized location which is currently located in Maradana. In Maradana Control Room, the train traffic controlling operations are divided into two units. They are Outstation and Metropolitan units. Outstation unit is for controlling trains running in outstation areas and the metropolitan unit is for trains running in Maradana, Coastal line and Metropolitan areas. Moreover, these sections have different underlying infrastructures such as mechanical interlocking and electro-mechanical interlocking based infrastructure.

The outstation control unit has mechanical interlocking based infrastructure which is the first generation of interlocking systems, the real-time location of the trains are visualized through the TDG (Time Distance Graph) which is manually drawn by the train controller to which the data provided by each station through FDM (Frequency Division Multiplexing) communication channel, when the train arrives and leaves the station. Controllers spend more time in plotting the TDG than decision making and for reporting purposes the same data are re-entered. This leads to more time and money consumption. Moreover, quick access to documents is a problem.

In the case of metropolitan unit, it has electro-mechanical interlocking based infrastructure which is the second generation of interlocking systems, the centralized system is directly connected to the interlocking system which drives the field equipments (point motor, signal lights, track circuits, gates, etc) and reports back to the system at the centralized control room. It has the system of manually operated switch boards in illumination based panel which is used to monitor and control train movements. This panel is very congested and the information is not very clear. For any failure in the panel, there is no backup mechanism. It is hardwired with the interlocking system resulting in thousands of complex wiring across the panel. The free space in the panel is not enough to add new tracks to the panel, which has become an extremely troublesome and time consuming task to the authorities. Availability of the system is reduced due to high time consumption in maintenance resulting in high down time of the system. Some parts of the system are obsolete and nowadays they are not manufactured commercially. Therefore the authorities are facing the burden of repairing the available equipment and custom based manufacturing of the devices is an extremely costly process.

Moreover, in both control units as a whole, the operation is entirely manual. All the reports and Train Graphs are generated by pen and paper and this has wasted a lot of time for the dispatcher's actual role of decision making. At the end of the day, all these problems will result in the trains getting delayed.

In this paper, an IT based solution named as CARCS (Computer Assisted Railway Control System) which addresses the aforementioned drawbacks and bottlenecks is presented. Johan Wikström, Arvid Kauppi, Arne W Andersson, and Bengt Sandblad have proposed the design principles of user interface for train traffic control [1]. Based on their user interface design principles, the proposed solution provides GUIs (Graphical User Interfaces) to monitor and control train movements using track layouts and

TDG, display train timetable, design track layouts and TDG and configure rules. McClanahan, R.H has mentioned about networked SCADA (Supervisory Control and Data Acquisition) architecture and its advantages [2]. The solution presented in this paper complies with the networked architecture in his inspiring paper. Earlier, A. L. A. T. D. Ambegoda, W. T. S. D. Silva, K. T. Hemachandra, T. N. Samarasinghe, and A. T. L. K. Samarasinghe proposed a Centralized Traffic Controlling System for Sri Lanka Railways which focuses only on the Metropolitan unit [3]. But the solution that is presented in this paper is a total solution for train traffic controlling in Sri Lanka. It has several added features and comes with high performance and more reliable hardware unit with higher I/O capacity. Moreover, all the standard hardware parts that were used to build the hardware unit are low cost and available in the market today.

II. Overall Architecture

The proposed system complies with the architecture depicted in Fig. 1. The overall architecture is a 4-tier networked architecture which is basically the distribution of system functionalities across network-connected systems. This architecture was chosen to implement the system because it fulfills the following non-functional requirements,

1) Performance: Since each tier is located on a physically separate computer, the processing power will be increased for the delivery of specific set of functionalities.

2) Maintainability: Because each tier is independent of the other tiers, updates or changes can be carried out without affecting the application as a whole.

3) Scalability: Because tiers are based on the deployment of layers, scaling out the application is reasonably straightforward.

4) Flexibility: Because each tier can be managed or scaled independently, flexibility is increased.

5) Availability: Applications can exploit the modular architecture of enabling systems using easily scalable components, which increases availability.

Typically, the proposed system belongs to the category of SCADA (Supervisory Control and Data Acquisition) systems. SCADA systems are used to monitor and control a plant or equipment in industries such as telecommunications, water and waste control, energy, oil and gas refining and transportation. The proposed system architecture complies with the current generation SCADA architecture [2]. The networked architecture is actually an enhanced version of distributed architecture since the functionalities are distributed across a WAN and not just a LAN. An advantage brought about by the distribution of functionality over a WAN is that of disaster survivability. By distributing the processing across physically separate locations, it becomes possible to build the system that can survive a total loss of any one location [4].

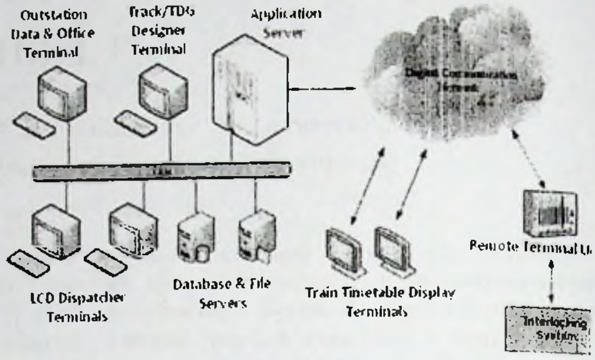

Figure 1. Overall architecture of the proposed system.

As illustrated in the Fig. 1, there are 4 tiers. They are,

1) Client: This tier comprises different terminals for different purposes. The identified client tiers are,

a) LCD Dispatcher Terminal: This is used by the dispatcher to monitor the movement of trains and make decisions. Particularly, DCV (Dispatcher Control View) is used to control trains in Metropolitan unit and TDG is used to control trains in Outstation unit.

b) Track/TDG Designer Terminal: This terminal is used by the administrators to reflect any changes that are made to the railway tracks/stations and it can be used to design a new track layout/ TDG.

c) Outstation Data Terminal: It provides an interface to the user to enter all necessary data which is used by the system throughout its operation.

d) Outstation Office Terminal: It provides the interface to enter the additional information on the train note. This additional information such as delay details and wagon positioning details are received after the train reaches its destination.

e) Train Timetable Display Terminal: This is intended for the delivery of dynamic train timetable information to the passengers who are at the railway station.

2) Application Server: This tier delivers the major functionalities of the system to its client tiers.

3) Database & File Servers: These will hold the central database and files such as track layouts and TDGs.

4) RTU (Remote Terminal Unit): This is the hardware interfacing unit which connects the proposed system with the existing railway interlocking system.

III. Track Designer and Dispatcher Control View

As the demand for the railway transport increases then there is a need to construct new rail tracks hence which will require it to be reflected in the centralized railway control system. The rail track designer is used to design the new track layouts or modify existing track layouts. The railroad designer can drag and drop the controls and components in the designer and set all the necessary properties of each control and save the layout.

C# .NET and WPF (Windows Presentation Foundation) are used to develop the track designer. WPF is used since it's a very rich platform and provides very rich graphic features. Moreover, WPF controls are inherently extensible by allowing for wide-spread customization of their appearance that adapts to the performance requirements [5]. Fig. 2 shows the screenshot of the track designer.

The designed track is saved in XAML (Extensible Application Markup Language) format. XAML excels in three key areas: expressivity, comprehensiveness, and extensibility of user interfaces [6]. Since XAML is not compiled, the user can edit it at anytime and quickly load it at runtime.

Once the designing of the track layout is finished, the XAML file is loaded in the DCV which is running in the client tier. Fig. 3 shows DCV with the track layout loaded. As mentioned earlier, this is the HMI (Human Machine Interface) that is exclusively used by train dispatchers for monitoring and controlling the train traffic in the Metropolitan unit. The HMI is a vital component of any SCADA system because it provides the interface to monitor and control the entire system. A SCADA system's accuracy and the stability depend mainly on the HMI. Moreover, all the GUIs (Graphical User Interfaces) are designed based on the guidelines on how to create a robust and an efficient user interface. The dispatchers are highly skilled users and the proposed GUI design is focused on giving them optimal support rather than being easy to learn [1].

Moreover, all the HMIs in the system are designed and developed giving more focus on the following requirements,

- Screen width and pixel limitations
- Fast response when updating the elements in the display.
- Ability to display the railway infrastructure with any level of detail.

In addition to the main features, the DCV has more features such as train no tracking, alarm generator with rules engine, generating standard set of reports, multi-user support, dynamic train timetable, traffic control playback (video log) and auto train supervision.

The rules engine comes with a Rules Editor which is a GUI that allows the administrator to design the workflow and define the rules as declarative conditions at runtime. In auto train supervision, the dispatchers can schedule the trains and their routes prior to their arrival so that when there is an oncoming train, the route is auto allocated. This enables to efficiently control trains that arrive/departure to/from a rail yard which has a very complex track layout. For example, the layout shown in Fig. 3 is a complex track layout where multiple route combinations are possible.

Figure 2. Track Designer

Figure 3. Dispatcher Control View

Traffic control playback is one of the notable features in the system which records all the current events and keeps it as a video log file to use it to see what has happened when something goes wrong in train controlling operations.

IV. TDG Designer and Time Distance Graph

TDG Designer is used to design TDG work sheet by adding or deleting stations in a railway line. Using the TDG Designer the administrator can select the stations to be appeared in the TDG work sheet and change the station labels to display the name of the station or to display the station code. The designed TDG work sheet will be used by the controllers in the TDG Application which is used to visualize the movement of each train running in outstation area added to the Train Note Component for a selected date. The Train Note was developed based on existing paper based train note that is being used by the controllers to enter the station code, arrival time, departure time and other details.

In this application the controller has to only enter the data in train notes of each running train and the TDG will be automatically generated whereas in the existing system the controller has to both enter the data and also draw the TDG. Event driven architecture principles were applied in developing the train note because the interaction between the TDG and Train Note is event based and also to make the two applications loosely coupled. Well adapted to the loosely coupled nature of distributed interaction in large-scale applications, the publish/subscribe communication paradigm has recently received increasing attention [7]. Fig. 4 depicts a screenshot of the Train Note and Time Distance Graph.

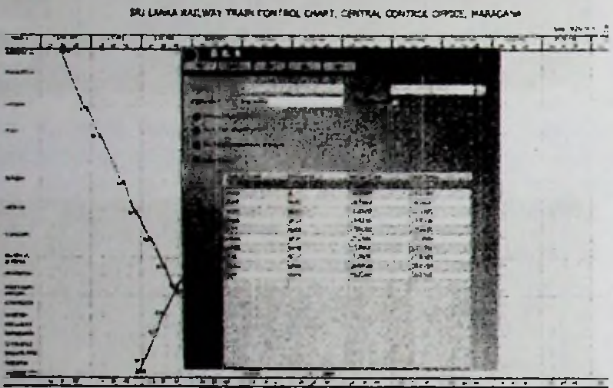

Figure 4. Train Note and Time Distance Graph.

V. REMOTE TERMINAL UNIT

This section explains the hardware interfacing unit of the proposed system. RTU (Remote Terminal Unit) is the hardware interfacing unit which connects the proposed system and the existing electro-mechanical relay based interlocking system. The interlocking system is a very basic safety system which purpose is to prevent trains from colliding and derailing. A basic interlocking system operates on the signals which guard each end of a segment (A segment is a partition of the railway track). By setting these signals appropriately, based on the locations of the involved trains, the interlocking system ensures that only one train has exclusive access to a segment [8]. Since the current interlocking system stays as one of the well established transport infrastructure of Sri Lanka, the proposed solution is fully focused on the centralized control room system which has many identified deficiencies as mentioned earlier.

In the current setting, the interlocking system is connected with the switch boards in illumination based panel directly. As mentioned earlier, this illumination based panel has two main functions, one is to display the indications that are reported from the interlocking system and the other is to send commands using switches (knobs, levers) to the interlocking system. Moreover, the indication signals come from indications relays of the interlocking system and the control commands are received by the control relays of the interlocking system. These indication and control relays are rated at 24V DC.

The proposed solution comprises a hardware unit which reads and controls the relays in the interlocking system and reports back to the system via TCP/IP communication link. The following challenges were faced when designing and implementing the hardware unit,

- Ability to support greater I/O capacity.
- Ability to send/receive signals quickly to/from the application server.
- Isolation from surge currents in the relay based interlocking system.
- Communicate with the application server via Ethernet interface.

The Fig. 5 illustrates the architecture of RTU of the proposed system.

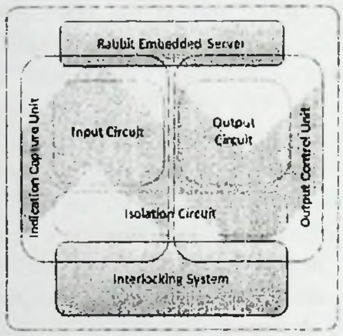

Figure 5. RTU Architecture.

As shown in the Fig. 5, the hardware unit was designed in a modularized way so that it is easy to maintain and troubleshoot. The controlling is done as follows, when the dispatcher gives command using the DCV, the application server receives the command and sends it to the target Rabbit embedded server in the required format (bit string). Then the Rabbit embedded server receives it via Ethernet port and sends it to the correct output card (circuit) which then will pick up the appropriate relay in the isolation circuit. Finally, that relay will pick up the relay at the interlocking system. Once the above process is finished, the interlocking system will carry out the rest of the controlling process based on its safety logic and reports back as an indication via indication relays which tells whether the command has been successfully carried out or not.

The indications are first received by the isolation circuit. Then it will be available at the input circuit. The embedded server pulls the indications which are at the input circuit and then pushes the indication using the correct format back to the application server. Finally it will be received by the DCV and displayed in LCD screen. The following section explains each of the modules of RTU in detail.

A. Rabbit Embedded Server

RCM3365 Rabbit Core was used as the embedded server which is shown in Fig. 6. The RCM3365 is based on a Rabbit 3000 microprocessor. The Rabbit 3000 is a high-performance microprocessor with low electromagnetic interference (EMI), and is designed specifically for embedded control, and communications. The 8-bit Rabbit 3000 outperforms most 16–bit processors without losing the efficiency of an 8–bit architecture. It is fast, running at up to 44.2MHz. So this satisfies one of the design challenges, ability to send/receive signals quickly to/from application server.

Referring to the memories, it has 512 KB flash memory, 512 KB program execution SRAM, 512 KB data SRAM and a 32 MB NAND Flash memory as well. The operating code data can be stored permanently into the flash memory using a Dynamic C FAT (File Allocation Table) file system., operating with a 1.8 V to 3.6 V power supply, the Rabbit 3000 boasts 52 parallel digital I/O lines. It's compact instruction set and high clock speeds give the Rabbit 3000 exceptionally fast math, logic, and I/O performance compared to other microprocessors [9]. Moreover, RCM3365 has 10/100Base-T RJ-45 Ethernet port which meets one of the design challenges of the hardware unit as well [10]. Ultimately, RCM3365 caters all the hardware requirements in a low cost and highly reliable way compared to other industrial controllers available in the market today.

B. Input/Output Circuits

The input/output circuits were built in order to increase the I/O capacity of RTU unit. The estimated numbers of I/O terminals for a particular section of the panel (for a relay house) were 101 control (output) and 256 indication (input) terminals. Likewise there are several sections (relay houses) in the centralized controlling which brings up one of the design challenges, supporting greater I/O capacity. Input and Output circuits are little bit different. They use different components in order to achieve its input/output capabilities. But their basic designs are same as shown in Fig. 7 i.e. both input and output circuits have the 8 channel bus which carries the signals to/from the microcontroller.

The input circuit has 8 octal bus transceivers (bi-directional buffers – 74HC245) and these buffers will allow data to be transferred in either direction between two busses. But the input circuit was designed so that the buffers work in one direction which transfer data received from isolation circuit to the 8 channel bus. The decoder is used to enable the buffer and make sure only one buffer writes to the bus at a time. Moreover, the microcontroller polls for data on the bus by controlling the decoder. Once the data are polled, Rabbit embedded server constructs the appropriate data string and sends it to the application server via Ethernet port. Since Rabbit microcontroller has 52 I/O pins, it is possible to connect almost 4 input circuits to the microcontroller which makes the input capacity of RTU to 256.

Figure 6. RCM3365 Rabbit Core Module.

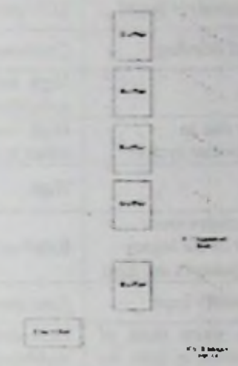

Figure 7. Basic design of the I/O Circuit.

The output circuit has same kind of basic structure, but it uses general purpose 8-bit storage registers instead of buffers. Because, in controlling process, the command is sent from microcontroller to isolation circuit which will then pick-up the relay at the interlocking system and these relays need to be kept energized for a certain amount of time period (20s) in order to carry out the rest of the process successfully but during this time period, the microcontroller must be able to send other commands in parallel.

This simultaneous operation is done using the storage register as follows, during the first command; the output of the storage register follows input. When the second command arrives to another storage register within 20s, the inputs at the bus are latched at the first storage register and now the output of the second storage register follows input at the 8 channel bus.

C. Isolation circuit

The main purpose of the isolation circuit is to provide the electrical isolation between the I/O circuit and the interlocking system. The isolation circuits are also little bit different for input and output. In the input isolation circuit, opto-couplers are used since the input terminals are less vulnerable to surge currents in the interlocking system. And the output circuits use miniature relays since the output terminals are more vulnerable to surge currents in the interlocking system. Moreover, output terminals are vulnerable to lightning strikes because they are connected directly to the relay house at the field whereas the input terminals are connected via an intermediate relay house near to the control room. Furthermore, proper methods of protecting the circuits and the system were taken into account [11-13].

VI. COMPARISON

There are lots of benefits in the proposed system compared to the existing system at the centralized control room as shown in Table 1.

TABLE I. COMPARISON BETWEEN EXISTING SYSTEM AND PROPOSED SYSTEM

	Existing System	Proposed System
DCV	Illumination based	LCD panel
TDG	Hand drawing	Computer generated
Reliability	Low	High reliability with the use of redundancy and backup servers.
Flexibility	Low due to hardwired system	High since the system can easily adapt to any change.
User friendliness	Low	High
Maintenance cost	Expensive since most of the legacy components are used	Relatively low cost.
Reporting	Manually kept	Auto generated
Throughput	Low since most of the work is done by the controller	Increased throughput using multithreaded execution and automation of the system.
Fault tolerance	Since it is hardwired, the failure at a point gets propagated.	Since the system is modularized, the fault isolation is achieved.

VII. SYSTEM TESTING AND RESULTS

Two onsite tests were carried out. The first test run was carried out at the Maradana Control Office of Sri Lanka Railways on 20th April 2010. A performance comparison based on the time taken to plot the time distance graph using the proposed system and the existing system was carried out. Fig. 8 shows the mean time taken for the controllers to enter train notes and draw the time distance graph at different time intervals during a day using both, existing system and proposed system.

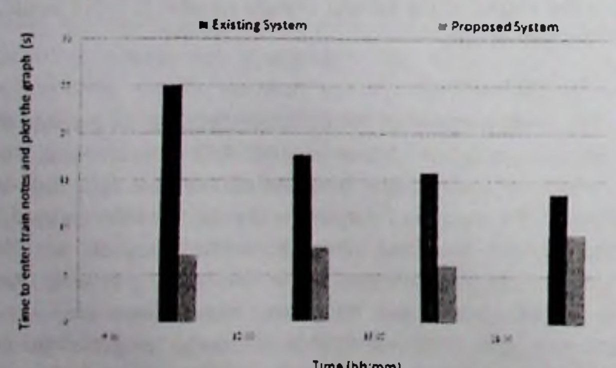

Figure 8. Time taken to enter train notes and draw the time distance graph using both, existing system and proposed system.

So from the results it is evident that the system has helped the controllers in outstation unit to carry out their work more efficiently than the existing system by reducing the time to note down the Train Notes and plot the Time Distance Graph by approximately 60%. All in all it has provided the controller an opportunity to spend more time on decision making.

A train simulator was developed which simulates the real train traffic scenario and the RTU. The full track design of three main stations was also designed and loaded using DCV in three consecutive LCD screens. Moreover, these stations were fully tested with the simulator prior to the second onsite test.

After the successful test with the simulation, the second test run with the real hardware unit (RTU) was done at Sri Lanka Railways workshop at Dematagoda on 8th May 2010 with the presence of senior officials of Sri Lanka Railways. The RTU was connected with their existing interlocking system. Then the track layout of metropolitan unit was configured with the proper I/O card addresses using the Track Designer interface. After the configuration, the DCV was loaded with track layout. The commands were successfully issued to the interlocking system and the indication signals were successfully captured. This proved the potential for actual implementation.

VIII. CONCLUSION

In this paper, we have presented a system which addresses the problems faced by the railway controllers of Sri Lanka Railways in carrying out the train traffic monitoring and controlling tasks. Features such as DCV, TDG, Track Designing, TDG Designing, Rules Editor, Report generation, Dynamic train timetable, Train Traffic Control Playback and Auto Train Supervision have helped to improve the overall operation of centralized train traffic controlling of Sri Lanka Railways. Especially, it has allowed the dispatchers to spend more time on decision making process. The presented onsite testing results provide evidence to this argument.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to our supervisors Dr. Chathura de Silva, Eng. A. T. L. P. Samarasinghe and Mr. Shanaka Ransiri for their tireless efforts and extended support towards developing the system. We also thank the Department of Computer Science and Engineering, University of Moratuwa and Sri Lanka Railways for giving us an opportunity to develop such a system and making this project a great success.

REFERENCES

[1] W. Johan, K. Arvid, and W. A. Arne, "Designing a graphical user interface for train traffic control," Uppsala University, Uppsala, Technical Report 2006-025, 2006.

[2] McClanahan, R.H., "The benefits of networked scada systems utilizing ip enabled networks," Rural Electric Power Conference, 2002. 2002 IEEE, 5-7 May 2002 Pages: C5 - C5_7

[3] A. L. A. T. D. Ambegoda. W. T. S. D. Silva, K. T. Hemachandra, T. N. Samarasinghe, and A. T. L. K. Samarasinghe, "Centralized traffic controlling system for Sri Lanka railways," in *ICIAFS08*. Colombo, 2008. pp. 145-149.

[4] Dale Barr, "National communications system," October, 2004. [Online].Available:http://www.ncs.gov/library/tech_bulletins/2004/tib_04-1.pdf. [Accessed: November 10, 2009].

[5] Kiran Kumar," WPF performance whitepaper," October, 2006 [Online] Available:http://blogs.msdn.com/kiranku/archive/2006/10/30/wpf-performance-whitepaper.aspx. [Accessed: September 20, 2009].

[6] J. Karsten and R. Jaime. (2007, Nov.) The new iteration: how xaml transforms the collaboration between designers and developers in wpf

[7] Patrick Th. Eugster, Pascal A. Felber, Rachid Guerraoui, Anne-Marie Kermarrec. "The many faces of publish/subscribe," *ACM Computing Surveys*, vol. 35, no. 2, pp. 114, June 2003.

[8] M. S. Madsen and M. M.Baek, "Modelling a distributed railway control system," *Informatics and Mathematical Modelling*. M.S thesis, Technical University of Denmark, DTU, 2005.

[9] Rabbit Semiconductor. (2002) Rabbit 3000 microprocessor user's manual. [Online]. http://www.rabbit.com

[10] Rabbit Semiconductor. (2005) RCM3365 rabbit core module user manual. [Online]. http://www.rabbit.com

[11] *IEEE Recommended Practice for Grounding of Industrial and Commercial Power Systems (Green Book)*, ANSI/IEEE Standard 142, 1982.

[12] *IEEE Recommended Practice for Powering and Grounding Sensitive Electronic Equipment (Emerald Book)*, ANSI/IEEE Standard 1100-1992, 1992.

[13] M. O. Durham and R. Durham, "Lightning, grounding, and protection for control systems," *IEEE Trans. Ind. Applicat.*, vol. 31, pp. 45–54, Jan./Feb. 1995.