INTERNATIONAL JOURNAL OF ELECTRICAL ENGINEERING & TECHNOLOGY (IJEET)

ISSN 0976 – 6545(Print)
ISSN 0976 – 6553(Online)
Volume 4, Issue 1, January- February (2013), pp. 75-80
© IAEME: www.iaeme.com/ijeet.asp
Journal Impact Factor (2012): 3.2031 (Calculated by GISI)
www.jifactor.com

IJEET
©IAEME

ANALYSIS AND IMPLEMENTATION OF LOCAL MODULAR SUPERVISORY CONTROL FOR A MANUFACTURING CELL - A CASE STUDY

S.C. Ramesh [a], Dr.G.Kalivarathan[b]

[a] Research Scholar, CMJ University, Meghalaya, Shillong.
[b] Principal/ PSN Institute of Technology and Science, Tirunelveli, Tamilnadu, Supervisor, CMJ University, Shillong. Email:sakthi_eswar@yahoo.com

ABSTRACT

In this Paper, a viable analysis and implementation of local modular Supervisory Control Theory for a manufacturing cell is predicted in the best possible manners to capture the necessities of idealizing the programmable logic controller (PLC). By means of local modular approach a methodology that reveals modularity of the plant and of the behavioral specifications models optimal supervisors are obtained without state-space explosion. For the purpose of simplifying the ladder diagram implementation in the PLC, reduced supervisors are implemented in a three level structure that executes the modular supervisor's concurrent action and interfaces the theoretical model with the real system. The flexible and productive manufacturing cell behavior, after the control system practical implementation, and the final PLC code readability and flexibility are positive quality indicators to the applied methodology.

Keywords: Programmable logic controller, Supervisory control theory, Discrete event systems, Supervisor generated controllable events

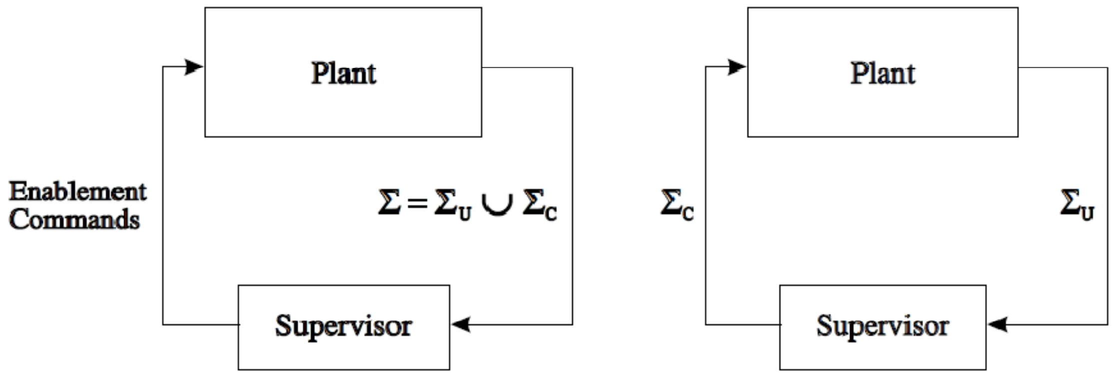
1. INTRODUCTION

With current technology, It is capable of building extremely large and complex systems. It is becoming increasing difficult to design controllers that guarantee the proper operation of such systems. For Discrete Event Systems, It enables us to design controllers for a given system that are mathematically guaranteed to be always controllable and to never deadlock. This is highly desirable in systems that are safety intensive. Our current work addresses the conversion of a theoretical supervisor to a physical implementation on a

Programmable Logic Controller (PLC). This has been investigated previously. To this end a testbed that physically simulates a manufacturing work cell was built around an Allen-Bradley PLC. In the remainder of this paper describe the modeling of the testbed, and the design of its controllers. Finally, we present several results that identify reduced plant models for verifying controllability and nonblocking. The Supervisory Control Theory (SCT), initiated by Ramadge and Wonham is a powerful framework for the synthesis of control for Discrete-Event Systems (DES). Although the SCT has obtained a wide acceptance in academy and some experimental applications have elucidated its potential, the extensive use of SCT in industry remains a promise. There are many reasons for the lack of industrial applications of SCT, including the computational complexity and state space explosion in the synthesis process. The number of states of the global system model grows exponentially with the number of subsystems. For that reason the complexity of computing optimal supervisors, although polynomial in the size of the global system, is a barrier for real problems that involve a large number of subsystems. Advances in computational hardware and complexity reduction techniques can help overcoming this barrier in many applications. However, the physical implementation of supervisors with a large number of states renders the control program immense, unreadable and, thus, untrustworthy. This approach reduces the computational complexity of the synthesis process and the size of supervisors by exploiting modularity of specifications and the decentralized structure of composite plants. Instead of a monolithic supervisor for the entire plant, a modular supervisor is obtained for each specification, taking into account only its local plant (affected subsystems). A necessary and sufficient condition named local modularity assures that the set of local modular supervisors has the same performance as the monolithic supervisor. As a consequence of the limited size of the resulting supervisors, the exponentially complex algorithm for optimal reduction of supervisors as becomes feasible. When local modularity holds, this approach has the advantage of providing more flexibility, computational efficiency and safety to the control application. This paper presents the results of an application of the local modular approach to the control program synthesis for a real manufacturing cell commanded by a programmable logic controller (PLC). Several new challenges and solutions arise from the process of linking the theory to reality, concerning aspects from the open-loop system modeling to the final structure of the control program. To guide the physical implementation of the control system from abstract supervisors, the authors propose a three level program structure that plays the set of reduced local modular supervisors concurrently, commands the evolution of decentralized sub plants and acts as an interface between the theoretical model and real control signals. This control structure differs from the implementation scheme proposed by other works for playing the supervisors exactly as previewed by the SCT while the interface levels resolve the differences between the abstract model and the real system. This characteristic allows a modular implementation of plant and of supervisors, what renders the control program clean and clear.

2. SUPERVISOR GENERATED CONTROLLABLE EVENTS

The standard RW theory model consists of a plant which spontaneously generates events, some controllable (can be disabled) and some uncontrollable, and a supervisor that tracks the plant's events, disabling controllable events according to some control law.



(a) (b)

Figure 1. (a) RW Feedback Loop (b) Balemi Feedback Loop

The plant is considered to be that which generates the events in the plant models. This includes the normal plant generator and the supervisor. In particular, it is not an important concern for the testbed since the plant model was designed to be flexible enough to allow for any contingency of the supervisor generating an event.

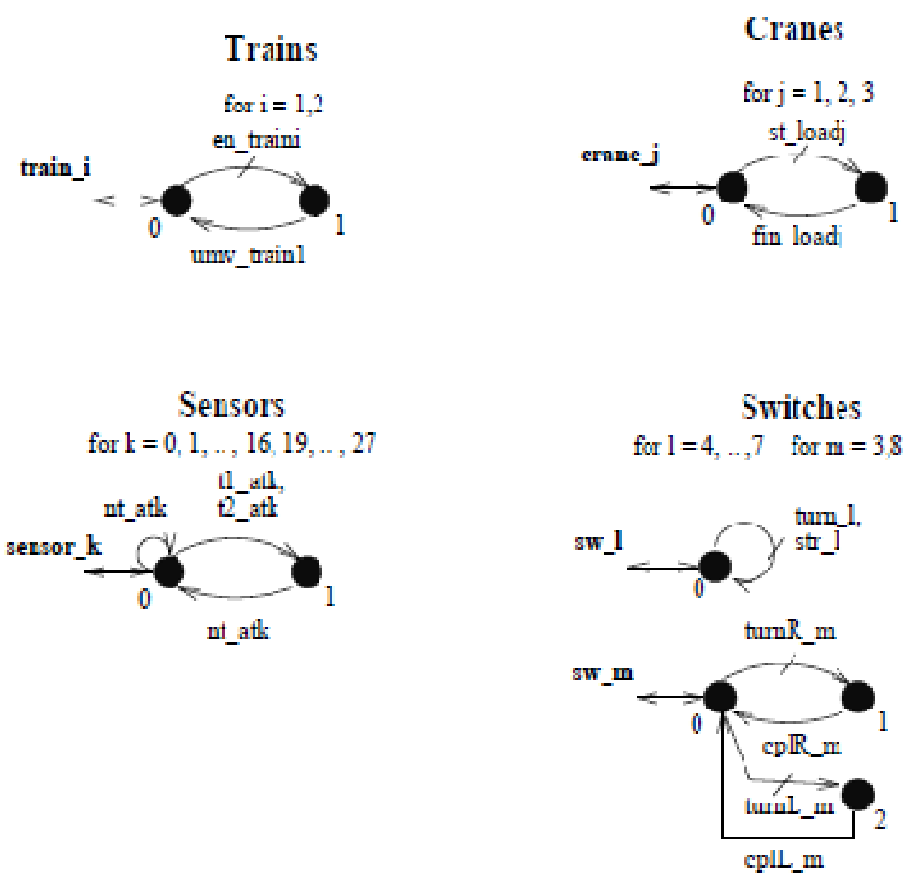


Figure 2. Plant Models

3. CRANE SUPERVISORS

The testbed's safety and operation specifications are given below. They are discussed in detail in the following subsections.

1. Prevent trains from colliding.
2. Ensure switches are set correctly while train traverses them.
3. Enforce specified route.
4. Synchronize trains to permit loading by cranes.

A crane supervisor continually monitors the sensor just in front of it for a train. When a train arrives, the supervisor stops the train and activates the crane, which then proceeds to load the train. When the crane finishes loading the train, the supervisor releases the train. In additional to feasibility issues, modular supervisors are easier (and more accurate in general) to design, modify, and implement. It permits one to easily convert the 25 supervisors discussed in this section into PLC code by hand. One single centralized supervisor would require computer software to do the conversion and the resulting PLC code would be far more massive than the modular code

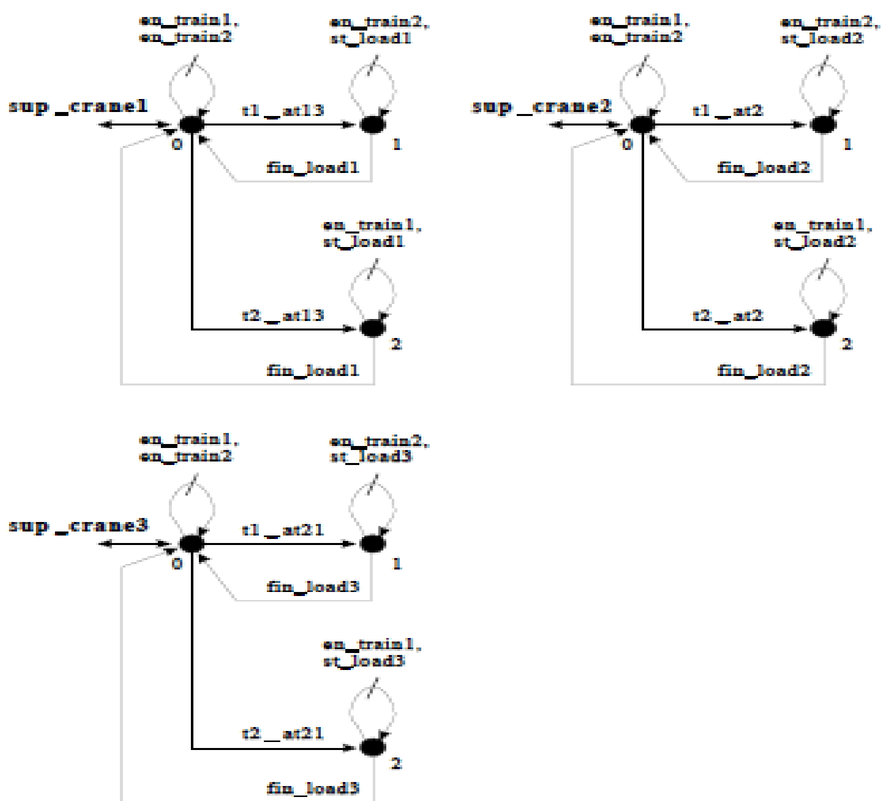


Figure 3. Crane Supervisor

4. ROUTING SUPERVISORS

The routing supervisors enforce the specified route of each train. Each train has its own set of supervisors, one for each switch used. The path of each train is controlled by the position of the switches as the train passes through them. The required pattern of switch settings for each train's route was found by tracing the path of each train. This overall pattern was then further reduced by observing the individual pattern of each switch. Finally, a supervisor was written for each switch to generate the prescribed pattern as the given train traverses the tracks. When all six supervisors for a given train are put together, the overall pattern creates the proper route for the train. Supervisors are the simplest form of supervisors, and are all very similar to the supervisor for track section RS23 and RS24. The layout of track section is RS23 and RS24. Sensor RS23 marks the start of the section, sensor RS26 marks the start of the adjoining section, and sensor RS22 marks the end of the preceding section. This causes complications, since a supervisor wants to interfere as little as possible with a train that plans to enter the other track, but the supervisor has no knowledge of the train's intended path.

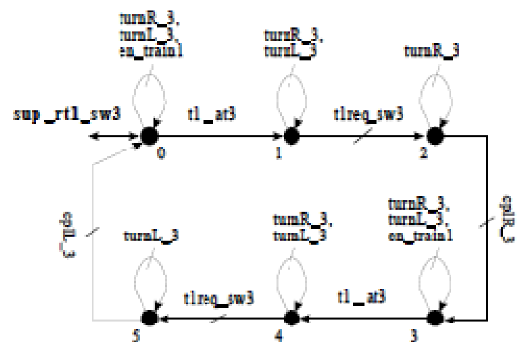


Figure 7.2: Routing Supervisor for Train 1, Switch 3

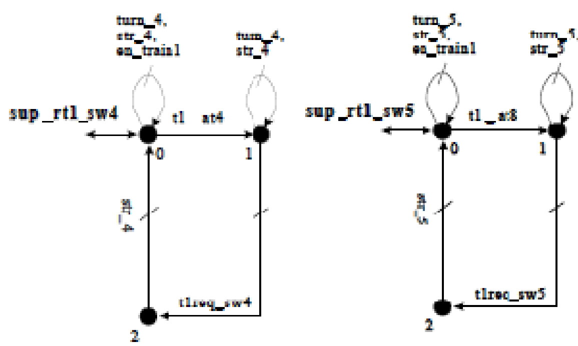


Figure 4. Routing Supervisor

5. CONCLUSION

The advantage of the testbed demonstrates that supervisory control can effectively be used as a design tool for reasonably large, real systems. It is noticed that modular design of both the plant model and the supervisor is essential to deal with large, real systems. The size of such systems currently makes any other method infeasible. The model reduction theorems greatly extend the possible size of plants that RW theory can be applied to. The theorems immensely reduce the amount of computations necessary. By considering the plant model as an interface specification, RW theory can easily be applied to programmable plants. The interface specification provides a necessary abstraction level between the high level DES supervisors and the low level software and hardware. The generic implementation methodology presented provides an excellent means to bridge the gap between DES theory and applications. By giving a physical interpretation to the theory, the implementation method makes it easier for designers to model real plants, and to design effective supervisors. Because the method is generic, it can be applied to a wide range of applications. Also, since the method uses CMSSM as an intermediate step, it can easily be modified for implementation on any digital logic device.

REFERENCES

[1] T. Catherall. A User's Guide to the MÄarklin Digital System. MÄarklin, Inc., 4th edition, 1991.
[2] Data I/O Corporation. ABEL-HDL Design Software, Language Reference, October 1991.
[3] W. Ford and W. Topp. MC68000, Assembly Language and Systems Programming. D.C. Heath and Company, 1987.
[4] Gebr. MÄarklin & Cie GmbH. MÄarklin Digital Control Unit 6021, 1993.
[5] D. Johnson. Programmable Controllers for Factory Automation. Marcel Dekker, 1987.
[6] R.J. Leduc and W.M. Wonham. Discrete event systems modeling and control of a manufacturing testbed. In Canadian Conference on Electrical and Computer Engineering, pages 793{796, Sept 1995.
[7] R.J. Leduc and W.M. Wonham. PLC implementation of a DES supervisor for a manufacturing testbed. In Thirty-third Annual Allerton Conference on Communication, Control, and Computing, Oct 1995.
[8] Yong Li. Supervisory control of real-time discrete-event systems. Master's thesis, Department of Electrical Engineering, University of Toronto, Toronto, ONT, 1986.
[9] G. Michel. Programmable Logic Controllers- Architecture and Applications. Wiley, 1990.
[10] M.P.O Morford and R.J. Lenardon. Classical Mythology. David McKay Company, 1976.
[11] Motorola, Inc. Motorola Freeware 32-bit Cross Assembler User's Manual.
[12] Motorola, Inc. MC68681 Dual Asynchronous Receiver/Transceiver, 1985.
[13] Motorola, Inc. MC68332 User's Manual, 1990.
[14] P. Ramadge and W.Wonham. Supervisory control of a class of discrete-event processes. SIAM J. Control Optim, 25(1):206{230, 1987.
[15] J. Stenerson. Fundamentals of Programmable Logic Controllers, Sensors, and Communications. Prentice Hall, Inc., 1993.
[16] R. Strobel and A. Johnson. Pocket pagers in lots of one. IEEE Spectrum, pages 29{32, Sept 1993.
[17] J. Wakerley. Digital Design Principles. Prentice-Hall, Inc., 1990.
[18] W. Wonham and P. Ramadge. On the supremal controllable sublanguage of a given language. SIAM J. Control Optim, 25(3):637{659, 1987.
[19] W.M. Wonham. Notes on Control of Discrete-Event Systems. Department of Electrical and Computer Engineering, University of Toronto, 1995.