行政院國家科學委員會專題研究計畫年度報告

具網路結合功能之半導體製造集結式機台子計畫一

集結式機台環境之發展及模式建立(III)

(The Modeling and Development of Cluster Tool Environment (III))

計畫編號：NSC 89-2218-E-002-0051
執行期限：89 年 8 月 1 日至 90 年 7 月 31 日
主持人：呂秀雄　黃漢邦　　研究人員：王哲龍
執行機構：國立台灣大學機械工程學系
Email: hanpang@ccms.ntu.edu.tw

中文摘要

最近幾年來，半導體製造在世界上蓬勃發展，晶圓的尺寸從八吋到變爲十二吋，使得單一晶圓製造越來越受重視。集結式機台乃針對單一晶圓製造而開發，其模組可以分爲晶圓移載模組，程序模組及卡匣式晶圓進出模組。

本研究提出集結式機台環境之發展與模型建立。首先利用分散式彩色時序裘式圖建立系統模型，然後並且開發集結式機台模擬器和控制器，同時將 SECS、HSMS 及 GEM 標準納入考慮，以設計智慧型模組控制器。除了近端控制之外，也利用網際網路設計遠端控制器達成遠端監控的目的。此外，爲達成品管檢測目的，亦發展統計製程管制模組。最後，利用電腦模擬驗証各系統模組。

關鍵詞：集結式加工機台、分散式彩色時序裘式圖、智慧型模組控制器、網路基底遠端控制器。

Abstract

A single wafer processing IC fabrication with flexible and distributed configuration is getting more important. The cluster tool is such an IC fabrication equipment. It consists of the cassette module, transport module, and process module. In this research, the model of a cluster tool is constructed using distributed colored timed Petri net (DCTPN). The emulator and controller of the cluster tool are further constructed. The standards of SECS, HSMS and GEM are used to develop the intelligent module controller. Finally, the web-based remote controller is developed and the SPC module is built in the cluster tool.

Keywords: cluster tool, distributed colored timed Petri net, intelligent module controller, web-based remote controller.

計畫緣由及目的

The IC fabrication industry plays an important role in the rapid growth of information technology. Moreover, the short product lifecycle, quick product change, and competitive market make the traditional batch-type production and fixed equipment IC fabrication insufficient to customerized market. A flexible, distributed configuration and single wafer processing IC fabrication is required. The cluster tool, as shown in Fig. 1, has multiple chambers with single-wafer processing. It is a key for the customerized market. In particular, the cluster tool can be operated in a small space and mini environment.

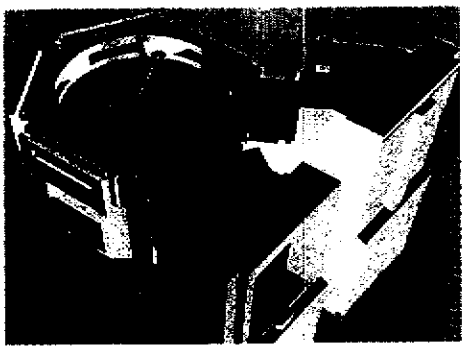

Fig. 1 Cluster tool platform

The cluster tool consists of the cassette module, transport module, and process module. Due to standardized control and mechanical interface, each module can be attached to a cluster tool and integrated into the cluster tool control system. Besides, the foundation of the cluster tool controller is the cluster tool module communication (CTMC) environment. The cluster tool controller and module controller can communicate over an Ethernet – TCP/IP network. However, the message transmitted from each controller is not in a standard format. The message format between the cluster tool controller and the module controller will be standardized using the intelligent module controller. The intelligent module controller employs agent technology to ensure message delivery and server loading balance. Based on the proposed intelligent module controller, the new coming module can be quickly configured and integrated into the system.

The virtual manufacturing (VM) is getting popular in the Internet era. It can provide a total solution for short lifecycle and customerized production. One way to achieve the virtual manufacturing is running the simulation model for the cluster tool. The distributed colored timed Petri net (DCTPN) will be proposed to model the behaviors of the cluster tool and run the simulation of the cluster tool.

On the other hand, the Internet-based manufacturing system is suitable for global and distributed manufacturing. The design, research and manufacturing need not be located at the same site. The web-based control is the key of Internet based manufacturing. A web-based control of the cluster tool will be proposed. Using the browser, the cluster tool can be controlled by the remote user.

研究方法

(1) DCTPN Model of Cluster Tool Controller and Emulator

Modeling of the cluster tool using DCTPN and the characteristic behavior of the cluster tool is great benefits. The emulator is designed and implemented using DCTPN.

1. DCTPN Introduction:

Carl Adam in 1962 proposed Petri net theory. Petri net can not only model a complex system in a graphic point of view but also have complete mathematical definition. In order to extend the capacity of the ordinary PN, Kuo [4] proposed the following extensions: color attributes, time property, distributed modular design, communication module, and arc multiplier. This new kind of Petri net is called distributed colored timed Petri net (DCTPN). A distributed hierarchical DCTPN is an eleven-tuple structure, DCTPN = (P_t, T_t, P_0, T_0, P_c, P_d, T_m, A, B, F, C), where P_t is a set of timed places; T_t is a set of timed transitions; P_0 is a set of immediate places; T_0 is a set of immediate transitions; P_c is a set of communication places; P_d is a set of COM server places; T_m is a set of macro transitions; A is a set of directed arcs; B is a set of inhibitor arcs; F is a set of interrupt arcs; C is the color set of transitions and places. The definitions and theorems of DCTPN are given in [3]. The icon definition of the DCTPN is shown in Fig. 2.

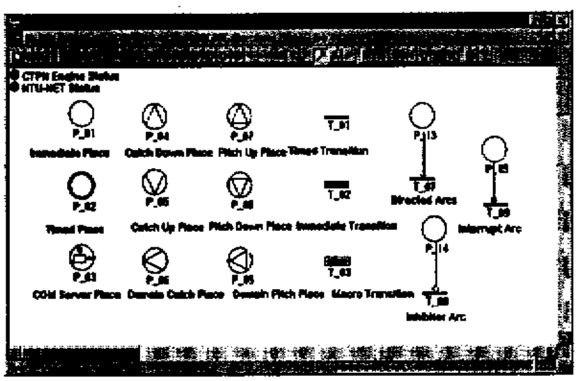

Fig. 2 Icon Definition of the DCTPN

2. Structure of a Cluster Tool:

A cluster tool consists of the cluster tool controller, transport module controller, process module controller, intelligent module controller, recipe management module, monitoring module, SPC module, in-situ metrology module, break down module, scheduling and dispatching module. Three process functions (RTCVD/Polysilicon, RTCVD/Nitride, and RTO) and one clean function are integrated into a cluster tool.

3. Modeling of a Cluster Tool:

DCTPN plays a major role to construct the cluster tool controller model. The intelligent module controller of a cluster tool can be modeled in terms of several individual modules. Each module consists of communication agent module. The cluster controller can be modeled in the superior DCTPN net. Based on the sequence diagram, the flow chart is easily constructed in Fig. 3 and can be transformed into the DCTPN net. Fig. 4 and Fig. 5 shows the DCTPN models of the cluster controller in the upper part and lower part. In this research, the intelligent module controllers of the process and transport modules are constructed. Fig. 6 shows the flow chart of the intelligent robot module controller, and the transport intelligent module controller is modeled as Fig. 7. The robot-scheduling program for either moving to destination position or returning to home position is determined by the place, P_15. The speeds of the robot movement, wafer pickup, and wafer place-down can be modeled by DCTPN subnets, including single end-effector robot (SEER) and dual-effector robot (DEER). The operation flow chart of the DEER and SEER robots are described in Fig. 8. The DCTPN models of DEER and SEER robot motions are shown in Fig. 9 and Fig. 10. The cluster tool cannot work when the key module, such as transport module, is broken or under maintenance. Therefore, the maintenance behavior and the fault diagnosis of the cluster tool modules, e.g. the robot module, cassette module, and process modules, must be included in the DCTPN model. The activities of the fault analysis and machine maintenance are modeled in the intelligent module level. The breakdown module is shown in Fig. 11. To analyze the failure modes and effects is an important thing to identify the various possible faults. For this reason, a fault tree plays a major role in solving the problems. Using a fault tree, the various possible failure modes and effects can be easily identified. For the robot module, most fault conditions can be divided into four cases including servo, peripheral, vacuum state and controller. The fault message will be sent to the cluster tool controller through NTUNet and the fault problem will be identified. After solving the fault problem, the message of finish repair will be issued, and the robot can work normally again.

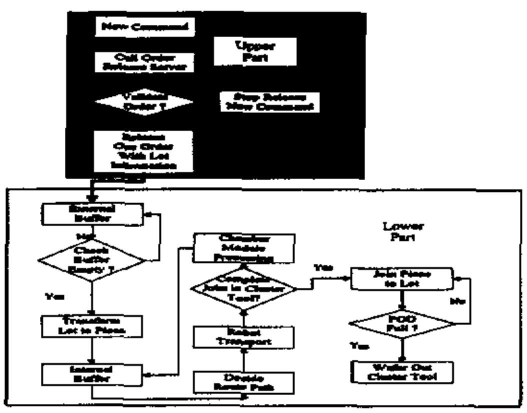

Fig. 3 Operation flow of the cluster controller

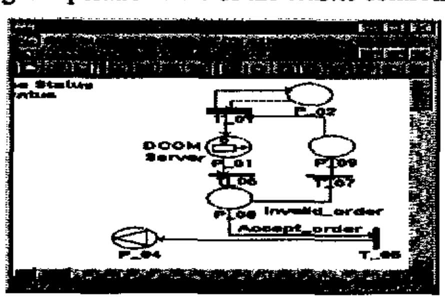

Fig. 4 Upper part of cluster tool controller

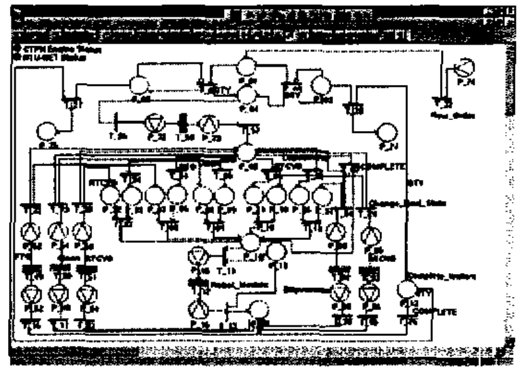

Fig. 5 Lower part of cluster tool controller

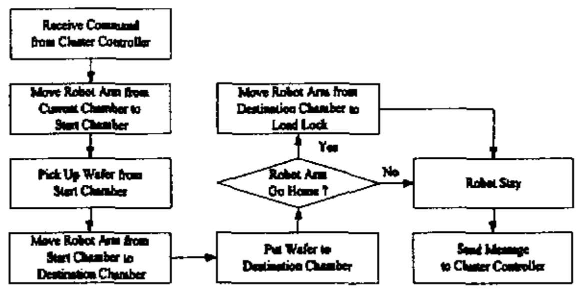

Fig. 6 Operation flow of an intelligent robot module controller

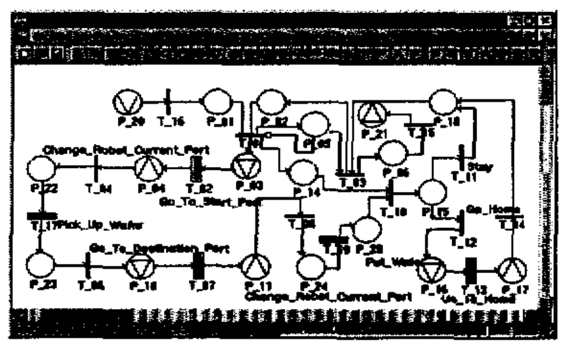

Fig. 7 Intelligent transport module controller

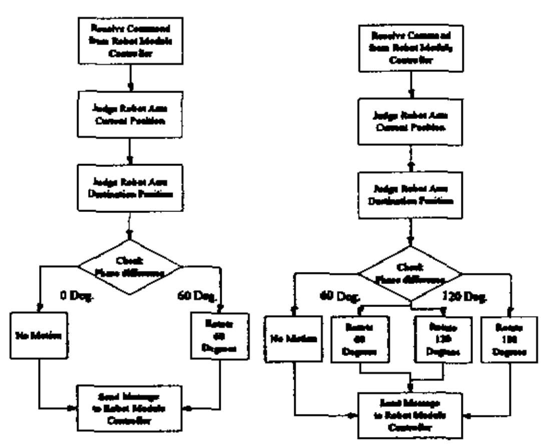

Fig. 8 The operation flow chart of the DEER and

SEER robots are described

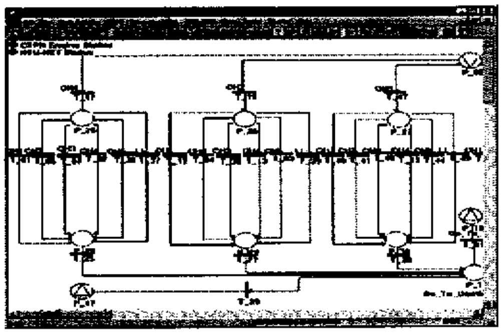

Fig. 9 DEER robot rotation model

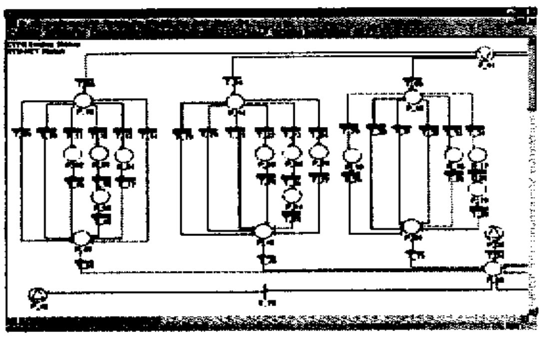

Fig. 10 SEER robot rotation model

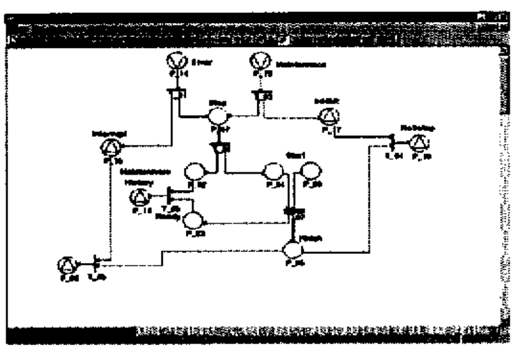

Fig. 11 The breakdown module

(2) Remote Control of the Cluster Tool

The remote control method will be introduced and the architecture of the control system will be discussed.

1. Remote control architecture and method:

The remote control application adopts three-tier architecture. The three-tier architecture includes front-end presentation, logical computation, and data layer. The front-end presentation layer shows data to the client. The logical computation layer stores/accesses data and perform mathematical operations. The data layer stores all kinds of information in the PC or workstation. The major advantage of the three-tier architecture is that every tier is independent. It is further elaborated below.

- Front End Presentation Layer: The Internet Explorer 5.01 is used as the browser.
- Logical Computation Layer: It may contain IIS, which can transform data to the form of HTML in the database using ASP and ActiveX control. Equipment (cluster tool) can use IIS to show the real-time data, sound and image on the remote monitoring system.
- Data Layer: SQL server can be used.

The monitoring computer momentarily checks the control state table of the server, and compares the control table of the monitoring computer. If there is no corresponding counterpart, the monitoring computer performs the mapping statement. The users in the remote computer can connect to the specific homepage on the specific server, and change the corresponding state table in the ASP program or ActiveX control program, such as the NetMeeting™ object. The monitoring computer transforms the statement in the state table into the standard language of equipment communication, such as GEM, SECS-II, and CTMC. Eventually, the message is sent to the cluster tool controller by the monitoring computer. Based on the NetMeeting™ object, which is the ActiveX control, the image and sound can be transferred from the factory to the remote side.

2. Web-based Remote Controller:

The web-based remote controller sends the status of the cluster tool and the on-site image and sound to the remote browser. The remote engineers can take action to the emergency and control the cluster tool controller in the factory. In order to get the information from the factory, the CCD cameras and web-based control are required. Therefore, the remote engineer can real-time control the cluster tool and the manager from the remote side can administrate the local operators in terms of critical condition. Using ASP technique, the login data can be checked and stored in the database of the server, as given in Fig. 12. After entering the controller, the user can manage the jobs of the cluster tool and run the cluster tool controller on the remote side through Internet. The server of the real-time controller, including ActiveX™ control objects, is implemented on the IISTM sever. The user from the remote side can launch the local control programs and fully operate the function of the cluster controller, as shown in Fig. 14. Due to the limited network bandwidth, there exists a time delay in transmitting images and sound. The serious time delay may block some alarms from receiving. In order to overcome this problem, the situation table of the cluster tool is constructed. Therefore, the text can be quickly transferred and the remote user is easily aware of the information from the cluster tool.

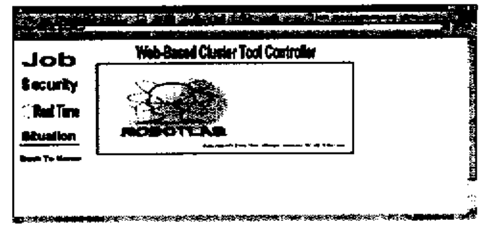

Fig. 12 The entry page of the web-based cluster tool

Fig. 13 Control from the remote side

(3) Design of the Intelligent Module Controller

The intelligent module controller will be constructed. The agent based module controller is developed using DCOM technology.

The cluster tool consists of process modules, transport modules and cassette modules. All modules communicate in CTMC environment and have many types of message flows. The intelligent module controller has to coordinate the six types of message flows. Using the agent technology, the intelligent module controller can ensure the loading balance and message delivery. The architecture of the intelligent module controller is flexible and can be easily reconfigured due to the characteristics of DCOM.

The intelligent module controller meets the CTMC standard and consists of control applications. The decision logic built in the control applications can control the status of the process module. The intelligent module controller can specify a set of SECS-II messages to support the CTMC service. Moreover, the agent technology is applied to the intelligent module controller and coordinates the message in the cluster tool controller. The module agent communicates with the cluster tool controller in the upper level and sends commands to the process module controller in the lower level, as shown in Fig. 14. The command format of the module agent is SECS-II. The operation procedure of the module agent is described as follows. First, the module agent identifies the cassette modules, transport modules and process modules. Second, the module agent connects to the database in order to obtain the data from the scenario table. After getting the data, the module agent determines the order of the process sequence. The way to find the order of the process sequence is to check the scenario identity and decide the scenario number, which indicates the beginning of the message flow. According to the flow chart, the module agent is developed using DCOM technology. Based on DCOM technology, the agent module is constructed as a DCOM server and executed on any registered server computers. Therefore, the DCOM server can achieve the loading balance of the server computers that run programs to control the process module.

結論與成果

The flexible cluster tool model was constructed in detail using DCTPN. The model can be easily reconfigured to fit in the real status of the cluster tool setup, including the robot module, process modules and breakdown module. Each module can incorporate with the breakdown module. The associated fault tree of the breakdown module is constructed by DCTPN. Based on the DCTPN modeling and simulation environment, the cluster tool can be easily reconfigured, modeled, controlled and evaluated. The cluster tool controller and web-based controller were developed. The computer in the remote side can easily control the cluster tool controller through intranet or internet. Other computers through the WWW browser can control the computer that installs the cluster tool controller. The intelligent module controller was constructed based on agent technology and DCOM technique. Using the agent technology, the intelligent module controller can ensure the loading balance and message delivery. The intelligent module controller communicates with the module emulator in terms of SECS-II protocol and GEM specification through HSMS or SECS-I. Finally, the model of SPC module was developed for quality control. The SPC server using DCOM technique can decide the control rules and take action to the status that is out of the control.

The model of the intelligent module controller is constructed using the DCTPN. The module agent is constructed as the DCOM server using Microsoft Visual C++ 6.0 and Active Template Library 3.0 (ATL). The user can use CTPNDesign program to setup the DCOM server and run the DCOM server on any server computer. The types of the DCOM server, including process module, robot module and cassette module, are chosen by the user.

The intelligent module controller

includes the NTUNet, ODBC driver, ActiveX control component and the module agent, as shown in Fig. 15. The intelligent module controller communicates with the process module based on SECS-II and GEM standards, which are AtiveX control of FASTech™. All scenarios of the process module and the intelligent module controller follow the scenario table and behavior table. In this work, SECSimPro™ or the process module simulator simulates the real process module. The process module emulator is constructed using FASTech™ ActiveX Control and modified according to the type of the process module. First, the message from the cluster tool controller is sent to the process module. Then, the process module retrieves the messages and handles the messages. Before processing the wafer, the recipe must be confirmed and loaded into the process module controller. If the setup of the process module is ready, the process module starts to process the wafer in the chamber as given in Fig. 16. The process module will retrieve the SECS-II messages and response it to process the wafer.

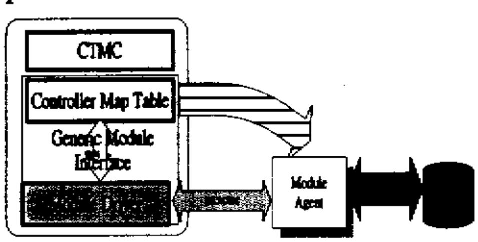

Fig. 14 The function of the module agent

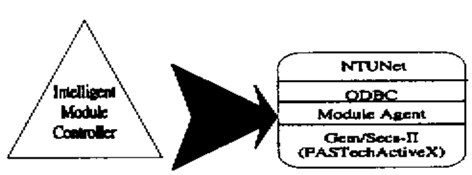

Fig. 15 The component of intelligent module controller

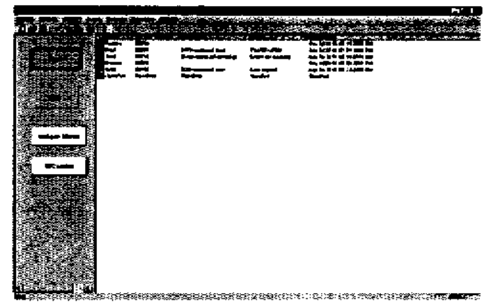

Fig. 16 Process module controller

參考文獻

[1] J.R. Moyne, H. Etemad and N. Najafi, "A Monitoring And Diagnostics System For A Plasma Etching Cell," *IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pp.48-53, 1991.

[2] G.H. Hu, Y.S. Wong and H.T. Loh, "An FMS Scheduling and Control Decision Support System Based on Generalised Stochastic Petri Net," *International Journal of Advanced Manufacturing Technology*, Vol.10, No.1, pp.52-58, 1995.

[3] H.P. Huang, and P.C. Chang, "Specification, Modeling and Control of a Flexible Manufacturing cell," *International Journal of Produce Research*, Vol.30, No.11, pp.1515-1543, 1989.

[4] C.H. Kuo, "Development of Distributed Component Based Manufacturing System Framewok," *Doctoral Dissertation*, Department of Mechanical Engineering National Taiwan University, 1999.

[5] C.H. Kuo, H.P. Huang and M.C. Yeh, "Object-Oriented Approach of MCTPN for Modeling Flexible Manufacturing Systems," *International Journal of Advanced Manufacturing Technology*,

Vol.14, No.2, pp.737-749, 1998.

[6] B. Lemmen, C. Van, H. Roede, J.E. Rooda, "Clustertool Optimization Through Scheduling Rule," *IEEE Proceeding of Semiconductor Manufacturing Conference*, pp.89-92, 1999.

[7] J.C. Moon, "An Event Channel-Based Embedded Software Architecture for Developing Telemetric and Teleoperation Systems on the WWW," *IEEE Proceeding of International Conference on Intelligent Robots and Systems*, pp.224-233, 1999.

[8] P.K. Mozumder, C.R. Shyamsundar, and A.J. Strojwas, "Statistical Control of VLSI Fabrication Processes: A Framework," *IEEE Transaction on Semiconductor Manufacturing*, Vol.1, No.2, pp.62-71, May 1988.