

# 반도체 생산 시스템의 자동 Socket 관리 기술 개발

°정 화 영 김 종 훈†

한남대학교 컴퓨터공학과 †제주교육대학교 컴퓨터교육학과

## Development of an Auto Socket Management Technique for Semicon Production System

Hwa-Young Jong Jong-Hoon Kim†

Hannam University †Cheju National University of Education

### 요 약

산업응용 시스템에 있어서의 생산성이란 시스템의 성능 및 효율성에 관한 척도가 된다. 따라서, 시스템의 개발 전과정에서 이를 향상시키려는 많은 노력이 이루어진다. 특히 반도체의 제조공정에서는 이를 위한 노력이 민감하게 발전되어왔다. 이들 중 IC Test Handler의 Socket관리는 제품(Device)의 생산량과 함께 시스템의 효율성에 직접적인 영향을 주는 부분으로 많은 연구 및 개발이 이루어지고 있다. 본 논문에서는 이러한 생산성 및 효율성을 높이기 위한 자동 Socket 관리 시스템을 개발하여 사용자(Operator)에게 보다 향상된 시스템 환경을 제공하고자 한다. 이를 위해 PC환경의 GUI 시스템을 도입하였으며, 실제적인 제어부분은 Real Time 운영체제를 탑재한 VME 시스템이 담당하였다. 또한, 신뢰성을 위하여 날짜별 데이터를 하드디스크에 저장하여 Socket 데이터를 사용자가 확인할 수 있도록 하였다.

### 1. 서 론

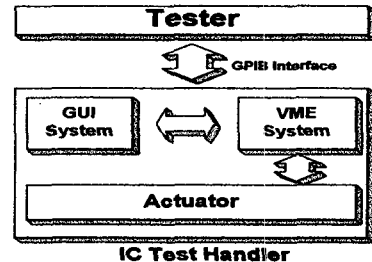
자동화기술의 도입에 있어서 시스템 제어기는 자동화 기기의 비용에 대한 중요한 요소를 나타낸다. 따라서, 많은 자동화 시스템에서 대부분 다양한 PLC(Programmable Logic Controller), VME, PC기반구조가 시스템 제어기로서 구현되고 있다[1]. 특히, 큰 변화중의 하나는 반도체 기술의 발달에 의한 PC(Personal Computer)사용을 들 수 있다. 또한, GUI(Graphical User Interface) 소프트웨어와 윈도우기반의 운영체제는 저렴한 개발비용효과와 신뢰성을 주었고, 오늘날 많은 제어 시스템들은 일반 PC환경에서 사용자에게 친숙한 GUI개발 소프트웨어 패키지에서 구현되고 있다. PC기반의 GUI시스템은 강력하고, 저렴하며, 사용자에게 의한 수정이 쉽다는 장점이 있다[2]. 따라서 자동화 시스템의 PC사용은 점차 확산되어가고 있으며 많은 연구 및 개발이 이루어지고 있다. IC Test Handler는 반도체 제조에 있어 후반부의 공정에 해당되는 장비이며 Handler의 Socket내에 Device(IC)를 삽입하여 적정온도 테스트 후의 결과로 불량 또는 불량제품을 분류하는 일을 담당한다. 따라서, Handler에서의 Socket 관리는 곧 생산량과 효율성을 결정하는 중요한 요소가 된다. 그러나, 많은 시스템이 이에 관한 정확한 규격이 없고, 실제 현장에서 Auto Socket관리에 관한 신뢰성 부족으로 이를 사용하지 않는 경우가 많다.

본 논문에서는 여러 가지 Socket Off 모드를 제공하여 다양한 Socket관리방법을 제시하여 반도체(IC)의 생산효율을 높일 수 있도록 하였으며, 사용자에게 쉬운 설정환경을 제공하고자 Windows 98환경의 GUI로 개발되었다. 또한 보다 정형화된 기준에 의한 관리방법으로 신뢰성 향상에 도움이 되도록 하였다.

### 2. 시스템 구성

#### 2.1 IC Test Handler의 구성

IC Test Handler는 여러 개의 Socket에 Device(IC)를 넣고 Tester에서 적정 조건에 따라 한꺼번에 Test한 후 그 결과에 따라 분류하는 자동화 시스템이다. 적정조건은 이미 정형화되어있어 이를 적용하여 결과를 얻는다. Handler에는 수많은 구동 기기가 있고, 그 흐름을 제어할 수 있는 사용자 환경(GUI)과 각각의 구동 기기를 제어하는 응용 부분이 있다. GUI부분은 PC에서 구현되었고 응용부분은 실시간 운영체제(VxWorks)를 탑재한 VME 시스템이 담당하였다. <그림 1>은 IC Test Handler의 시스템 구성을 나타낸다.

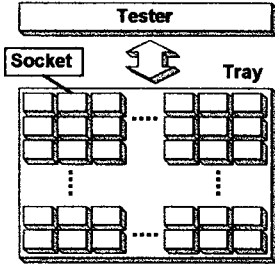


<그림 1> IC Test Handler 시스템 구성

#### 2.2 Socket의 운용

IC Test Handler내의 Tray(Device를 이송하는 기구)는 실제 Device를 넣고 Handler내의 여러 부분을 이송하는 역할을 담당한다. 또한, Tray내에 Device를 넣는 자리를 Socket이라 한다. 이 과정에서 Tester는 Socket내의 Device를 적정

조건에 따라 Test하게 되고, 그 결과에 따라 Handler는 Socket내의 Device를 자동으로 분류하는 역할을 담당한다. 따라서, Tray내의 Socket에 대하여 Device 적재량이 많을수록 Handler의 생산효율은 높아지게 되나, 적재량이 많아도 사용할 수 없는 Socket이 많을수록 생산효율은 떨어지게 된다. 즉, Socket의 효율적인 관리 및 운용은 Handler의 생산효율을 높일 수 있는 중요한 요인이 된다. Socket을 사용할 수 없게되는 가장 큰 요인은 기구적인 문제를 들 수 있다. 이는, Socket이 Tester와의 접속시 기구적으로 잘 맞지 않는 문제에서 비롯된다. <그림 2>는 Socket의 구성을 나타낸다.



<그림 2> Socket의 구성

많은 Handler 제조업체들은 자사의 자동화 시스템의 생산효율을 높이기 위해 Socket관리에 관한 많은 방법들을 제시하여왔다.

### 3. Auto Socket Off 관리 시스템 개발

#### 3.1 Socket Yield 산출

본 논문에서 Socket에 관한 Yield는 다음과 같은 방식으로 산출하였다.

$$\text{Socket} [\#\text{Num}] \text{의 Total} = \text{Socket} [\#\text{Num}].\text{Cat}[i_1] + \text{Socket} [\#\text{Num}].\text{Cat}[i_2] + \text{Socket} [\#\text{Num}].\text{Cat}[i_3] + \text{Socket} [\#\text{Num}].\text{Cat}[i_4] + \text{Socket} [\#\text{Num}].\text{Cat}[i_5] + \text{Socket} [\#\text{Num}].\text{Cat}[i_6] + \text{Socket} [\#\text{Num}].\text{Cat}[i_7] + \text{Socket} [\#\text{Num}].\text{Cat}[i_8]$$

( 단, #Num : 1 ~ 64,

$i_1 \sim i_8$  : 결과 Category Number

Socket [#Num].Cat [ ] : 해당 Socket에 대한 Category의 Device Count )

$$\text{Socket} [\#\text{Num}] \text{의 Good} = \text{Socket} [\#\text{Num}] \text{의 Cat}[1] + \text{Socket} [\#\text{Num}] \text{의 Cat}[2] + \text{Socket} [\#\text{Num}] \text{의 Cat}[3].$$

( Category 1, 2, 3 를 Good 으로 Setting시 ).

$$\text{Socket} [\#\text{Num}] \text{의 Yield} = (\text{Socket} [\#\text{Num}] \text{의 Good} \cdot 100) / \text{Socket} [\#\text{Num}] \text{의 Total}.$$

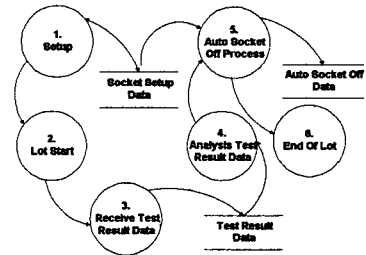
#### 3.2 Auto Socket Off관리에 관한 방법론

Handler운용 중 자동으로 Socket을 관리하기 위하여 본 시스템은 2가지의 적용모드와 5가지 사용방법론 및 한가지 Socket관리모드를 제시한다. 적용모드 중의 첫째는, Lot 적용이다. 이는, 운용 중 Auto Socket Off가 일어나도 다음 물량이 시작되면 Off된 Socket을 모두 사용 가능하게 한다. 즉, 다음물량의 시작시 Auto Socket Off에 관한 사항을 초기화하여 처음부터 다시 적용하도록 한다. 둘째는, 항상 적용이다. 다음 물량이 시작되더라도 Auto Socket Off에 관한 사항은 변함없이 진행되며, Off된 Socket도 초기화하지 않는다. 또한, 적용Yield도 Lot에 관한 Socket Yield가 아닌 누적

Yield가 기준이 된다. 사용방법에 있어서는 첫째로, 절대평가 방법을 들 수 있으며 이는, 해당 Socket Yield를 기준으로 Socket을 관리한다. 절대평가시의 Yield를 사용자가 설정하면 Handler운용 중에 각 Socket별 Yield를 산출하여 설정 Yield이하에 해당되는 Socket은 모두 Socket Off시킨다. 둘째로, 상대평가 방법이다. 사용자가 적용Yield를 설정하면 Handler 운용 중 전체평균 Yield를 산출하고, 그 수치에 적용Yield를 감한 Yield가 기준이 되어 그 이하의 Socket을 모두 Off시킨다. 셋째로, 연속평가 방법으로 기준은 Yield가 아닌 설정 Category가 된다. 즉, 사용자가 적용 Category와 적용횟수를 설정하면 운용 중에 Test결과가 연속적으로 적용횟수이상 적용Category가 발생했을 경우 해당 Socket을 Off시킨다. 넷째로, 연속 + 상대평가 방법이며, 마지막으로 연속 + 절대평가 방법이다. 이는, 연속평가방법과 상대평가 방법을 혼합한 것과 연속평가방법과 절대평가방법을 혼합한 방법으로 두 가지 방법 중 먼저 적용대상이 되는 Socket을 Off시킨다. Socket관리모드는 Socket Disable 기능으로 이는 사용자가 적용횟수를 설정하면 Auto Socket Off가 적용횟수 이상 발생시 이를 사용자가 임의로 Socket을 Off시킨 효과를 낸다. 즉, 일정횟수이상 Auto Socket Off가 발생했을 경우 이는 사용자가 직접수리를 하여야 하는 경우로 간주하여, 수리가 되기 전까지 Socket을 Off시킨다. 따라서, Lot 적용 시에도 Socket을 초기화하지 않는다.

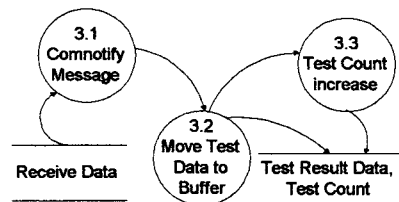
#### 3.3 구현

Auto Socket에 관련한 프로세스에서 Socket Off 부분은 5 번째에 해당된다. 또한, 하나의 공정이 끝나는 시점인 End Of Lot의 경우엔 사용방법과 적용모드를 검사하여 Socket의 초기화 여부를 결정하여야 한다. <그림 3>은 Auto Socket Off에 관한 전체 처리부분의 DFD를 나타낸다.



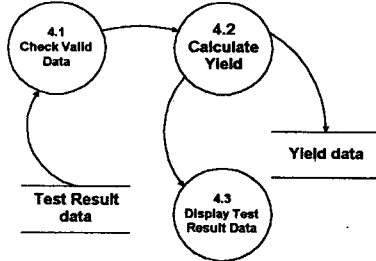
<그림 3> Auto Socket Off 처리

<그림 3>에서의 3번째 항목에는 Test된 결과정보를 VME 시스템으로부터 전송 받는다. 따라서, 적용시점을 결정하기 위하여 Test Count를 증가시키고 전송된 결과정보는 Buffer에 놓는다. <그림 4>는 Test 결과를 VME 시스템으로부터 전송받는 부분이다.



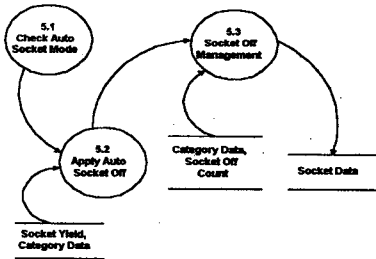
<그림 4> Receive Test Result Data DFD

Buffer로 옮겨진 결과정보는 인식 가능한 결과수치인지를 검사하는 유효성 검사를 받는다. 유효한 정보인 경우 이를 근거로 각 Socket별 Yield와 전체 평균 Yield를 산출하게 된다. 다음 <그림 5>는 결과정보를 분석하는 부분의 DFD를 나타낸다.



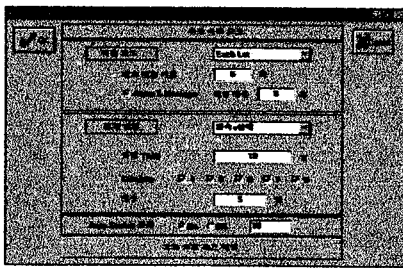
<그림 5> Analysis Test Result Data DFD

산출된 Test Count, Socket Yield, 평균 Yield등의 수치에 따라 Auto Socket Off기능을 수행한다. 따라서 현재의 적용방법, 적용모드 및 관리모드를 확인하여 Socket Off기능을 결정한다. 또한, Socket Off된 횟수를 계산하여 적용 Category에 따라 Socket을 영구히 Off시킨다. 다음 <그림 6>의 Auto Socket Off 처리에 관한 프로세스는 이를 나타낸 것이다.



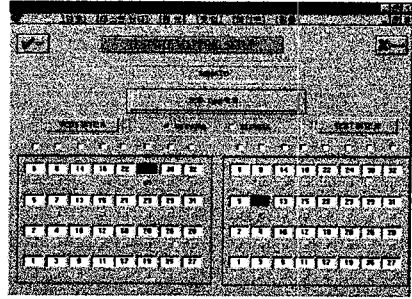
<그림 6> Auto Socket Off Process DFD

이를 구현한 시스템 화면을 <그림 7>에 나타내었다. 이는, Auto Socket의 적용에 관한 초기 설정화면으로, 사용자가 적용모드 및 방법을 설정하게되면 각 설정치에 따라 Auto Socket Off가 동작하게 된다.



<그림 7> Auto Socket Off 설정화면

또한, Socket의 설정상태를 비교하여 사용자가 임의로 Off시킨 Socket에 대하여는 Auto Socket의 결과에 관계없이 적용하지 않는다. 따라서, <그림 8>은 사용자가 임의로 Socket을 관리할 수 있는 Socket 설정화면을 나타낸다.



<그림 8> Socket 설정화면

Auto Socket Off 적용이후 사용자는 전체 Socket의 상황에 대한 정보를 확인할 수 있어야 하고 이를 관리할 수 있어야 한다. 따라서, 사용자가 임의로 Off시킨 Socket을 포함하여 Socket 전반에 관한 상태를 각 Yield 및 수량으로 나타낸다.

#### 4. 결론

Socket의 자동관리는 사용자에게 시스템의 효율성 및 생산성을 증가시키는 중요한 요인이 된다. 따라서, 본 시스템에서는 다양한 적용방법 및 기법을 제공함으로써 보다 효율적인 Socket관리를 수행할 수 있었다. 또한 항상적용모드에서는 누적 yield를 적용하여 지속적인 Socket의 관리를 행할 수 있었고, 사용자가 이를 초기화할 때까지 적용되었다. 특히, Socket관리모드에서 Socket Disable을 On으로 설정시 설정 횟수 이상으로 Auto Socket Off가 발생하면 해당 Socket을 Off시킨 다음, 이후에 발생하는 Auto Socket Off기능에서 제외함으로써 사용자에게 Socket의 상태를 점검하게 할 수 있었다. 또한, 일반 사용자에게 익숙한 PC환경의 Windows기반으로 구성되었다는 면에서 초보자라도 쉽게 Socket에 관련된 설정을 할 수 있었다. 그러나, 본 시스템에서 제공되어지는 Socket의 정보를 체계화하여 데이터베이스로 구성된다면 사용자에게 보다 많은 History 데이터를 제공할 수 있을 것이다. 또한, 많은 적용방법 및 기법들을 사용자의 운용실정에 맞게 간략화 한다면 보다 사용자에게 친화적인 시스템을 구축할 수 있을 것이다. 제조공정을 담당하는 시스템은 생산성 및 효율성에 따라 운용여부를 결정할 만큼 중요한 비중을 차지하고 있다. 따라서, 이를 자동화하여 시스템의 성능을 향상시키려는 노력은 지속적으로 이루어져야하며 연구되어야 할 것이다.

#### 참고문헌

- [1] R. L. Anderson, J. M. Reagin, T. D. Garner, T. E. Sweeny, "Open-Architecture Controller Solution for Custom Machine Systems", SPIE, Vol. 29, No. 12, 1997.
- [2] Kevin Borthwick, Pardip Thind, Philip Franssen, "PC-Based Operator Interface", IEEE Industry Application, Vol. 4, No. 4, July/August, 1998.