

# 클러스터 컨트롤러

The Cluster Controller

이해문(H. M. Lee)  
정태진(T. J. Chung)

산업기술지도실 선임연구원  
산업기술지도실 책임연구원, 실장

반도체 산업에서는 디바이스 제조 비용의 절감과 정밀도 향상을 위한 공장자동화의 한 방향으로 '80년대 후반부터 여러 공정의 반도체 제조장비를 클러스터화 하여 국제적 표준으로 제정하려는 노력이 꾸준히 진행되어 왔다. 반도체 장비의 클러스터 표준화는 제조 메이커가 각기 상이한 장비의 호환성과 상호 운용성을 증대시킴으로 장비 고유의 생산성과 디바이스 제작 능력의 한계를 확장시키고 제조기간과 비용을 크게 절약시킬 수 있다. 본 고에서는 클러스터 툴이라고 하는 반도체 장비의 핵심기술인 클러스터 컨트롤러에서의 모듈간 통신 및 제어 기술을 중심으로 국제적 표준화활동의 진행과 기술의 발전방향에 대하여 소개하고자 한다.

## 1. 개요

클러스터 툴이란 반도체 디바이스 제조의 성능과 생산성 향상을 위해 하나의 웨이퍼 핸들러에 2개 이상의 모듈화된 진공 챔버를 부착하여 프로세스 시스템을 병렬로 수행하는 멀티챔버 프로세싱 제조장비를 의미한다. 반도체 산업에서 이와 같은 멀티챔버 프로세싱의 개념은 반도체장비의 자동화 분야에서 부분적으로 도입되어 널리 사용되어 왔으나, 최근에 디바이스가 더욱 복잡하고 회로선 폭이 미세하게 발전하면서 클러스터 툴에서의 멀티챔버 프로세싱 기술의 장점이 중요하게 부각되고 있다.

반도체 산업에서 클러스터 툴을 사용하는 이유는 다음과 같다. 우선 클러스터 툴과 이에 상응하

는 단일 웨이퍼 툴의 집합과 비교하여, 클러스터 툴은 점유 공간이 작으면서도 같거나 훨씬 높은 처리능력(throughput)을 제공한다. 그리고 단일 웨이퍼 툴에서 독립적으로 수행되는 각각의 프로세스 스텝들을 하나의 툴에서 연속적으로 수행함으로써 클리닝 스텝과 같은 중간 단계가 생략될 수 있으며, 웨이퍼의 로드(load)가 전체 공정의 수행중에 한번으로 완료되기 때문에 단일 툴 사이의 웨이퍼 이동 및 수작업에 의한 웨이퍼 로드와 언로드(unload) 과정에서 발생할 수 있는 웨이퍼 손상과 대기노출에 의한 오염 등을 제거할 수 있다.

또한, 클러스터 툴의 프로세스 챔버 각각은 상호 교환될 수 있도록 표준화된 방법으로 모듈화되어 있어서 독립적인 단일 웨이퍼 툴 기능을 가질 수 있으며, 하나 이상의 모듈들이 사용자가 원하

는 형태로 교체되어 사용될 수 있기 때문에 최상의 유연성과 확장성을 제공한다.

그러나 클러스터 틀은 멀티챔버 프로세싱을 사용하기 때문에 단일 웨이퍼 틀보다 크게 복잡하며, 각각의 멀티챔버들은 독립적인 개스 주입 및 진공 시스템을 갖고 있어 전체적인 유지보수 시간을 증가시킨다. 그리고 부가적인 클리닝 스텝이 거의 없기 때문에 첫번째 프로세스에 로드되는 웨이퍼의 청결도 유지와 챔버 내의 진공 상태 유지는 단일 웨이퍼 틀보다 더욱 엄격하게 모니터링되고 제어되어야 한다. 또한 중앙의 웨이퍼 핸들러를 공유하는 각각의 챔버가 수행하는 프로세스들이 상호 연관성을 갖도록 공조되기 위한 클러스터 틀 아키텍처는 프로세스간의 상이한 여러 이슈들을 수용할 수 있도록 보다 세심한 설계가 요구된다.

## II. 클러스터 틀 아키텍처

클러스터 틀은 크게 플랫폼이라고 하는 클러스터 틀 하드웨어와 이를 운용하고 관리하는 클러스터 컨트롤러(소프트웨어)로 구분할 수 있으며 본고에서는 반도체 장비 자동화의 핵심기술인 소프트웨어 부분을 중심으로 설명하고자 한다.

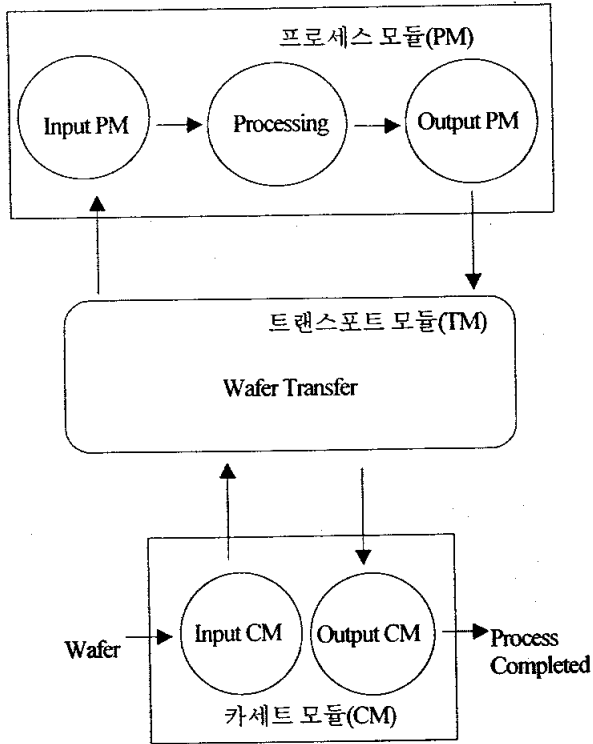
### 1. 클러스터 틀 하드웨어

클러스터 틀은 하나의 장비에서 다수의 독립적인 웨이퍼 프로세스 스텝들을 자동화에 의해 연속적으로 처리할 수 있는 구조를 가져야 한다. 또한 이기종간의 상호 운용성과 저비용의 유지보수를 보장하기 위해, (그림 1)과 같은 클러스터 틀 내부

에서의 웨이퍼의 이동 및 처리 모델에 따라 상호 독립적인 처리기능을 갖는 클러스터 모듈들로 나누어 진다. 클러스터 틀은 기능에 따라 카세트 모듈(cassette module: CM), 트랜스포트 모듈(transport module: TM), 프로세스 모듈(Process Module: PM)의 조합으로 구성되며, 국제적인 표준규격으로 모듈 간 접속 인터페이스를 규정하여 플러그 앤 플레이 방식으로 쉽게 접속할 수 있도록 제공된다. 각 클러스터 모듈들의 기능 및 운용절차는 다음과 같이 구분된다.

- 카세트 모듈: 클러스터 틀의 웨이퍼의 입력과 출력을 담당하는 모듈
- 트랜스포트 모듈: 웨이퍼 운송 로봇(웨이퍼 핸들러)에 의해 카세트 모듈과 트랜스포트 모듈, 트랜스포트 모듈과 프로세스 모듈 사이의 웨이퍼 운송을 담당한다.
- 프로세스 모듈: 실질적인 웨이퍼 프로세싱을 담당하는 모듈로서 서로 다른 처리절차를 갖는 여러가지 프로세스 모듈들이 플러그 앤 플레이 방식으로 쉽게 접속될 수 있다.

클러스터 틀은 카세트 모듈의 입력포트에 입력된 웨이퍼를 트랜스포트 모듈의 웨이퍼 핸들러에 의해 프로세스 모듈로 반송하고 프로세스 모듈에서 레시피(recipe)라고 하는 웨이퍼의 프로세싱 방법과 순서를 규정한 공정 시퀀스에 의해 정해진 프로세스 스텝들을 거쳐 프로세싱이 완료된 웨이퍼는 다시 로봇 핸들러에 의해 카세트 모듈의 출력포트로 반송하면 일련의 클러스터 틀 시퀀스가 완료된다

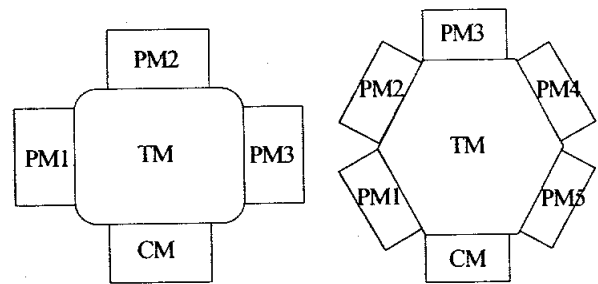


(그림 1) 클러스터 모듈간 웨이퍼 운송모델

클러스터 툴 하드웨어의 물리적인 구조는(그림 2)와 같이 일반적으로 4각형과 6각형 구조가 사용되고 있다. 기본적으로는 트랜스포트 모듈을 중심으로 1개의 카세트 모듈과 수개의 프로세스 모듈이 부착된 구조를 취한다. 각 프로세스 모듈은 독립적인 단일 웨이퍼 툴로서 사용이 가능하지만 클러스터 툴의 효율을 최대화 하기 위해서는 클러스터 툴이 적용되는 반도체 제조공정에 최적화되도록 설계되어야 한다. 예를 들어 5개의 독립된 웨이퍼 프로세싱을 거쳐 완성되는 제조공정을 통합하여 하나의 클러스터 툴에서 수행할 때는 경우에 따라 프로세스 스텝 수를 2개 이내로 줄일 수 있도록 설계될 수 있다.

또한 클러스터 툴의 모든 프로세스 모듈의 처리

능력은 동일하게 매칭되어 각 프로세스 모듈이 웨이퍼 가공을 위해 대기하는 일이 없도록 하여야 한다. 즉 모든 프로세스들은 동일한 진공레벨에서 처리되고 모듈간의 처리시간도 유사하게 설계되어, 각 프로세스 모듈이 웨이퍼를 제시간에 핸들러에 인도하여 다음 스텝으로 갈 수 있도록 하여야 한다.

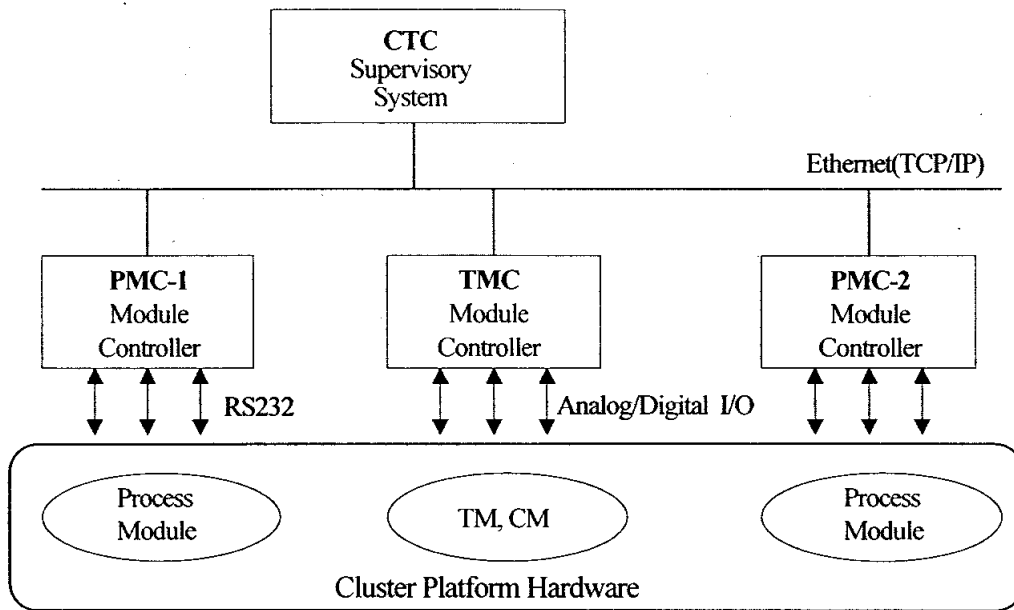


(그림 2) 클러스터 툴 하드웨어의 기본구조

## 2. 클러스터 컨트롤러

클러스터 컨트롤러는 모듈화된 플랫폼 하드웨어를 관리하고 제어하는 운용시스템으로(그림 3)과 같이 CTC(cluster tool controller)라고 하는 하나의 호스트 컴퓨터, TMC(transport module controller), PMC(process module controller) 등의 모듈 컨트롤러가 LAN으로 연결되어 있으며, 이들 운용시스템 전부를 클러스터 컨트롤러라 한다. CTC는 전반적인 클러스터 툴 운영상태를 모니터링하고 TMC와 PMC 모듈 컨트롤러를 제어하며, TMC는 트랜스포트 모듈과 카세트 모듈의 운영과 제어를, 그리고 PMC는 프로세스 모듈의 운영과 제어를 담당한다.

클러스터 컨트롤러는 주된 기능으로 웨이퍼 반송에 관한 제어, 모듈 내에서의 웨이퍼 프로세스



(그림 3) 클러스터 톨 소프트웨어(클러스터 콘트롤러)의 구성

관리, 그리고 클러스터 톨 외부로부터의 웨이퍼 입출력 처리 기능을 수행한다. 그리고 주기성을 수행하기 위한 보조기능으로 레시피 관리, 예외 처리, 데이터 리포팅 등의 처리기능을 수행한다

가. 웨이퍼 입출력 처리

클러스터 톨은 외부로부터 카세트 모듈의 입출력 포트를 통해 캐리어(carrier)의 형태로 웨이퍼가 이동된다. 캐리어는 통상적으로 20개 정도의 웨이퍼 그룹을 의미하며, 캐리어 내의 한 웨이퍼를 슬롯(slot)이라 한다. 동일한 캐리어 내에서 슬롯별로 웨이퍼 프로세싱은 다를 수 있으며 또한 슬롯내에 웨이퍼가 없는 경우도 있으므로, 클러스터 콘트롤러는 이들 슬롯을 각기 구분하여 톨 내에서의 웨이퍼 위치 및 이동경로 등을 관리하여야 한다. 웨이퍼 입출력 처리기능은 다음과 같이 캐리어 교환과 캐리어 매핑(carrier mapping)으로 구분한다.

1) 캐리어 교환

캐리어 교환은 클러스터 톨 외부로부터 카세트 모듈로의 웨이퍼 입력과 출력을 의미한다. 클러스터 콘트롤러는 웨이퍼 프로세싱이 완료되어 캐리어 교환이 요구될 때 CM으로 입출력 트랜스퍼 명령이 발급될 수 있도록 스케줄링이 이루어진다. 캐리어 교환은 SMIF(standard mechanical interface) 캐리어, AGV(automated guided vehicles), 혹은 오퍼레이터의 수작업 등 여러가지 방법이 사용되고 있으며 CM 입출력 트랜스퍼 명령에는 CM이 정확하게 캐리어 교환을 제어할 수 있도록 캐리어 교환 방법과 캐리어 내의 웨이퍼 정보 등이 자세히 정의된다.

2) 캐리어 매핑

카세트 모듈은 클러스터 톨 외부로부터 캐리어를 받으면 각 슬롯마다 웨이퍼의 존재여부 확인과 웨이퍼 ID를 배정하여 클러스터 콘트롤러에 보고

하여야 한다. 그리고 클러스터 콘트롤러는 카세트 모듈로부터 얻은 웨이퍼 정보가 프로세스 모듈에서 수행될 배치 프로세스의 레시피와 일치하는지를 확인하여야 한다. 또한 클러스터 콘트롤러는 배치 프로세스가 진행될 때 캐리어로부터 이동되는 웨이퍼의 순서와 이동된 웨이퍼들의 클러스터 틀 내에서의 현재 위치에 관한 정보를 모니터링한다.

#### 나. 웨이퍼 프로세스 관리

프로세스 모듈은 레시피에서 슬롯별로 정의한 웨이퍼 프로세싱 순서와 방법으로 프로세스 잡 (process job)을 처리한다. 이 때 프로세스 잡의 범위는 웨이퍼가 프로세스 모듈에 도착하기 전의 개스 주입, 챔버의 진공유지 등 프로세스 모듈 전처리 과정과 프로세싱 이후의 후처리 과정이 포함된다.

#### 다. 웨이퍼 반송제어

웨이퍼 반송제어는 카세트 모듈과 프로세스 모듈, 프로세스 모듈과 프로세스 모듈 사이의 이동을 의미하고, 모든 이동은 트랜스포트 모듈의 웨이퍼 핸들러라고 하는 웨이퍼 반송 로봇에 의해 수행된다. 클러스터 콘트롤러는 웨이퍼를 이동시키기 위해 다음과 같은 절차에 의해 관련된 모듈을 적절하게 준비시켜야 한다.

- 웨이퍼를 트랜스포트 모듈로 보내기 위한 출발지 모듈의 출력포트에 웨이퍼를 이동시키고 모듈간의 진공상태를 평형으로 유지하며 격리 밸브를 연다.
- 트랜스포트 모듈의 웨이퍼 핸들러가 출발지 모듈의 포트에서 웨이퍼를 집어온 후 출발지 모듈의 격리 밸브를 닫는다.

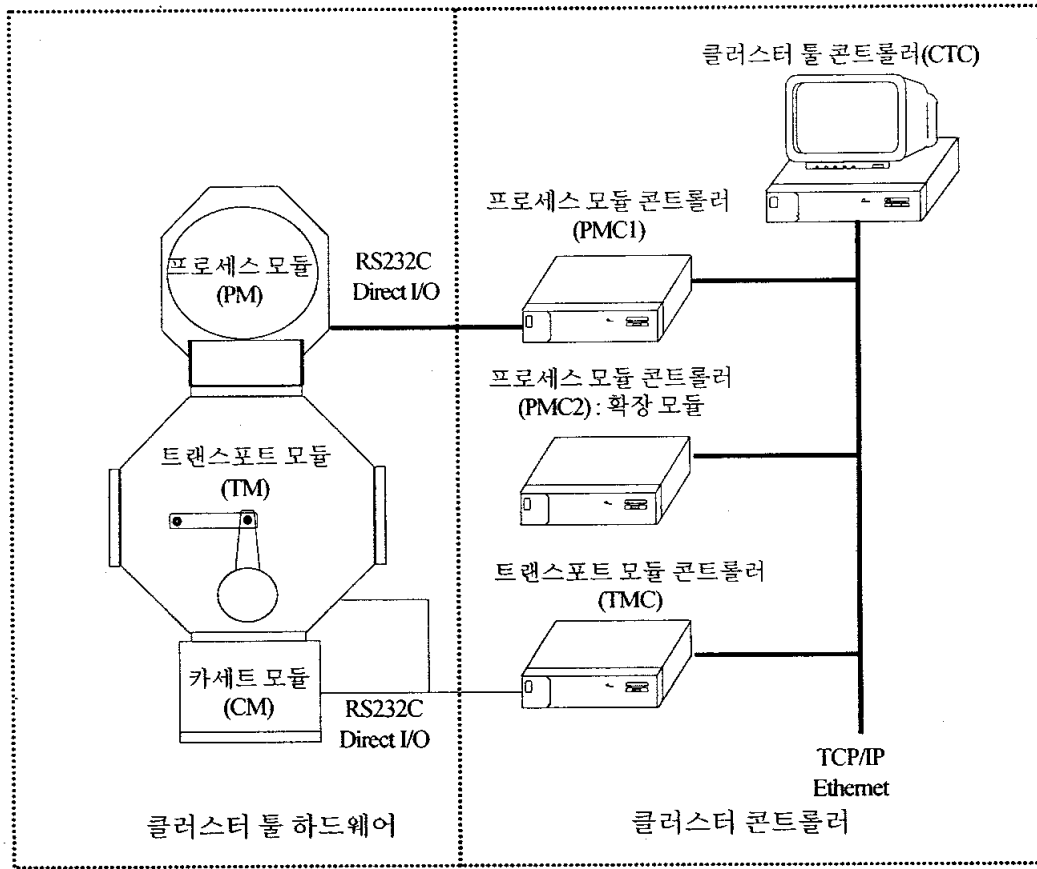
- 목적지 모듈의 격리 밸브를 열고 웨이퍼 핸들러로부터 목적지 모듈의 입력포트에 웨이퍼를 뱉은 후 격리 밸브를 닫는다.

#### 라. 예외 처리

클러스터 툴의 오퍼레이터 인터페이스는 일반적으로 모듈 콘트롤러로부터 원격지에 있게 되므로 에러 복구를 위한 예외처리 기능은 매우 필수적이다. 이러한 예외 처리는 모듈 일부의 하드웨어 failure와 같이 시스템이 자체적으로 해결할 수 없는 경우의 회복이 가능한 비정상적 상태가 예외 조건으로 정의된다. 이 때의 처리결정에 관한 권한은 오퍼레이터에게 있으며 클러스터 콘트롤러는 이미 정의된 예외 조건을 만났을 때 이의 해결을 위한 오퍼레이터 decision의 입력을 요구하며, 오퍼레이터는 abort, retry 등의 적절한 복구 액션을 결정하여 수행토록 한다.

#### 마. 레시피 관리

레시피는 PM에서 수행될 프로세싱에 관한 여러가지 파라미터들을 사용자가 지정한 정보 파일로서 레시피 헤더와 바디(body)로 구성된다. 클러스터 콘트롤러는 CTC를 통해 사용자가 레시피를 작성하고 편집할 수 있는 기능과 이미 작성된 레시피들을 선택하여 프로세싱을 위한 레시피 업로드 혹은 다운로드 기능을 제공한다. 프로세스 모듈의 레시피 처리기는 CTC에 의해 요구받은 레시피의 내용이 논리적으로 타당한가를 검사하며, 레시피의 처리 내용을 해석하여 프로세스 잡을 수행한다.



(그림 4) 클러스터 툴의 전체적인 구조

바. 데이터 리포팅

데이터 리포팅은 클러스터 모듈에 관한 각종 변수값을 수집하여 보고하는 기능으로 클러스터 컨트롤러는 이벤트(event) 리포팅과 트레이스(trace) 리포팅의 2가지 유형의 데이터 리포팅을 제공한다. 이벤트 리포팅에서는 각 이벤트의 타입과 수집되어야 할 데이터를 정의한 이벤트 리포트 정의(event report definitions)를 제공하며, 클러스터 모듈은 이벤트가 발생했을 때 이벤트 리포트 정의에 따라 관련된 데이터를 수집하여 리포트 메시지를 발생시킨다. 트레이스 리포팅은 프로세싱 조건의 변화와 같은 중요한 데이터들을 이벤트가 발생한 순

간부터 일정한 시간 간격으로 데이터를 수집하여 보고하는 기능으로, 데이터 수집에 관한 사항을 트레이스 리포트 정의(trace report definitions)에서 규정한다.

3. 클러스터 컨트롤러의 네트워크 구조

클러스터 모듈 컨트롤러간 접속방식은 시리얼(serial) 접속에 의한 point-to-point 방식과 LAN 접속에 의한 네트워킹 방식이 사용되고 있다. 과거에는 경제적인 이유로 시리얼 접속방식이 많이 사용되었으나 OSI(open system interconnection) 표준방식을

따르는 클러스터 모듈 통신표준이 상용화되면서 통신속도와 Fab 자동화에 대한 이점 때문에 TCP/IP를 기반으로 하는 Ethernet의 사용이 일반화되고 있다. (그림 4)에 클러스터 모듈의 네트워킹을 포함한 클러스터 툴의 전체적인 구조를 예시하였다.

### III. 클러스터 표준화 동향

미국을 중심으로 하는 반도체장비 제조업체들은 클러스터 툴의 개발비용을 최소화하고 더욱 저비용의 제품을 생산하며 보다 공개된 시장을 창출하려는 목적으로 1989년 3월 MESA(modular equipment standard architecture)라는 표준화 그룹을 결성하였다. MESA는 반도체산업의 국제적 표준화 조직인 SEMI(semiconductor equipments materials international)에 의해 공식적인 인증이 주어졌으며, 1989년 9월 SEMI의 소속조직으로 명칭을 MESC(modular equipment standards committee)로 변경하였다. MESC는 클러스터 툴 관련 모든 표준화 활동을 주도하여, '94년까지 150mm와 200mm 웨이퍼 프로세싱을 위한 모듈간 물리적 인터페이스와 제어통신에 관한 여러 표준을 완성하여 클러스터 툴 표준의 중요한 기반을 제공하였다. 현재 MESC의 공식적인 활동은 중단된 상태이지만 SEMI 내의 여러 표준위원회에서 300mm 웨이퍼 프로세싱을 위한 클러스터 툴 표준과 클러스터 툴을 기반으로 하는 반도체 Fab 자동화에 관한 표준이 활발하게 진행되고 있다.

#### 1. SEMI/MESC 표준

MESC 표준은 앞서 언급한 바와 같이 클러스터

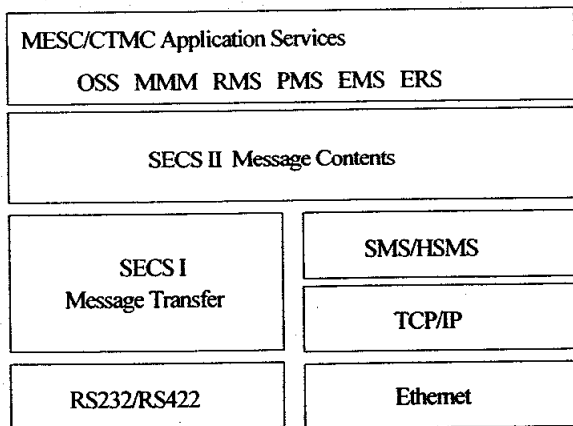
툴의 중요한 기반이 되는 표준으로 '94년에 완성되어 발표된 이후 현재에는 많은 상업용 응용제품들이 사용되고 있다. MESC는 여러 장비 공급자들로부터 제공되는 클러스터 모듈의 상호 운용성이 보장되는 최적의 클러스터 툴 표준을 성공적으로 개발하기 위해 많은 부분을 SEMATECH(semiconductor manufacturing technology)의 도움을 받아 완성하였다. SEMATECH는 미국의 반도체 제조사가 중심이 된 비영리 연구단체로서 '80년대 초반부터 반도체 공정자동화를 위한 통신모델인 GEM(generic equipment model) 표준을 개발해오고 있었으며, MESC는 클러스터 툴이 소규모의 공정자동화 장비라는 개념하에 모듈간 통신 프로토콜의 하부구조로서 GEM 표준의 SECS-I, II 프로토콜을 채택하고 있다.

MESC 표준은 크게 하드웨어와 소프트웨어 부분으로 나누어진다. 하드웨어는 클러스터 모듈간의 물리적 접속규격을 정의하며, 소프트웨어는 하드웨어의 운영과 제어를 위한 모듈간 통신표준으로 구성된다. 하드웨어 표준은 SEMI E20부터 E26까지 6개의 표준과 부록으로 구성되어 있으며 클러스터 툴 제작에 소요되는 구성부품간의 접속을 규격화함으로써 다수업체들로부터 공급되는 부품들을 조합하여 가장 저렴하고 우수한 기능을 갖춘 저비용의 솔루션을 제공하려는 업체간의 경쟁을 유도하고 있다. 또한 반도체장비 메이커 이외에 제 3자의 부품공급을 활성화 하여 클러스터 툴 사용자의 부품 확보가 용이하도록 하는 동시에 부품의 재고와 유지보수 비용을 줄일 수 있도록 하는 방향으로 표준화가 제정되었다.

클러스터 툴의 운용 및 모듈간 통신을 정의하는

소프트웨어 표준은 (그림 5)와 같이 반도체 공장자동화 표준인 GEM/SECS-II의 애플리케이션인 CTMC(cluster tool module communications) 표준을 기반으로 하며, MESC는 클러스터 표준이 다음과 같은 기능을 가질 수 있도록 CTMC를 개발하였다.

- 단순화(scalability): 복잡한 시스템을 단순화된 모듈로 구성하여 정교한 솔루션과 저비용의 구현을 제공한다.
- 성능(performance): 처리능력을 최대로 하기 위해 제어/통신에 소요되는 overhead를 최소로 최적화한다.
- 신뢰성(reliability): robust/reliable한 제어를 가능하게 하는 통신은 매우 어렵지만 여러회복 기능을 통해 그러한 가능성을 제공한다.
- 호환성(alignment): 다른 표준들과의 호환성을 제공하여 유연한 통합이 가능하도록 한다.



(그림 5) MESC 표준서비스 및 프로토콜 구조

#### 가. SECS-I

SECS-I(message transfer)은 호스트 컴퓨터와 반도체 장비간의 메시지 전송방식을 정의하는 프로토콜로서 physical, block, message, transmission의 4가지

논리적 계층으로 이루어진다.

- Physical 계층: RS232C 케이블을 이용한 물리적인 접속 계층
- Block 계층: SECS 메시지의 기본 단위인 블록의 구조와 포맷에 관한 규정
- Message 계층: 하나 이상의 블록으로 구성되는 메시지의 전송에 관한 규정
- Transmission 계층: 질의 메시지와 응답 메시지의 송수신 방법에 관한 규정

#### 나. SECS-II

SECS-II(message contents)는 SECS-I을 통하여 교환되는 메시지의 내용에 관한 세부사항으로 반도체 제조공정에서 대표적으로 요구되는 활동(activity)만을 표준 메시지로 정의한다. SECS-II에서 제공하는 활동으로는 제어프로그램의 전송, 웨이퍼 이동 정보, 측정 및 시험데이터, 경보(alarm) 등에 관한 사항을 규정하고 있다.

#### 다. SMS

SMS(SECS message service)는 서비스 사용자 사이의 상호작용에 관한 모델, 서비스의 액션과 이벤트의 순서 관계 등을 정의하여, SECS 메시지를 OSI의 애플리케이션 계층과 프레젠테이션 계층의 환경에서 사용할 수 있도록 서비스 정의와 프로토콜을 규정한다.

#### 라. HSMS

HSMS(high-speed SECS message service)는 TCP/IP 네트워크에서 SECS-II 메시지의 전송계층을 규정하는 표준으로 '95년 5월에 최초로 발표되었다. 또한 HSMS는 하나의 호스트 시스템이 다수의 장비



를 동시에 접속하여 제어할 수 있도록 제공한다. 이는 물리계층으로 TCP/IP LAN을 사용하여 RS232보다 훨씬 빠른 데이터전송을 제공하기 때문에 SECS-I과 SMS의 기능을 대체하여 폭넓게 사용되고 있다.

마. CTMC

CTMC(cluster tool module communications)는 클러스터 툴의 전반적인 운영과 관련하여 모듈간 통신의 개념 및 방법을 정의하는 표준으로 MESC 표준의 골격을 제공하는 중요한 표준이다. CTMC는 클러스터 툴의 애플리케이션 개발과 기본적 요구기능에 관하여 다음에 설명하는 6개의 서비스 기능으로 정의하며, 서비스 각각의 세부사항은 독립된 SEMI 표준들로 규정되어 있다.

1) OSS

MESC 표준은 OMT(object modeling technique) 기법을 사용하여 모듈간 통신을 표현하고 있으므로 OSS(object services standard)에서 object에 대한 공통된 개념, 절차 및 서비스 등을 정의하여 MESC의 다른 표준에 대한 레퍼런스(reference)를 제공한다.

2) MMM

MMM(material movement management standard)은 자동화된 웨이퍼 반송에 관한 표준으로 클러스터 툴 내에서의 웨이퍼 반송의 개념, 관련 디바이스의 운용절차, 웨이퍼 반송을 수행하기 위해 필요한 메시지 서비스 등에 관한 세부사항을 정의한다.

3) RMS

RMS(recipe management standard)는 CTC와 클러스터 모듈간 레시피 관리의 통합을 위한 개념, 절차 및 서비스를 정의하며, 클러스터 툴이 정확하게

웨이퍼를 프로세싱할 수 있도록 하는 레시피의 전송과 관리에 관한 서비스를 제공한다.

4) PMS

PMS(processing management standard)는 캐리어 내 웨이퍼 그룹의 동시적인 프로세싱과 캐리어 슬롯의 개별적인 웨이퍼의 동일하고 반복적인 프로세싱을 수행하는 job의 관리에 대한 개념, 운용절차, 메시지 서비스 등을 정의한다.

<표 1> 클러스터 툴 소프트웨어 MESC 표준의 주요내용

SEMI E4	SEMI equipment communications standard I (SECS I)
SEMI E5	SEMI equipment communications standard II (SECS II)
SEMI E13	SECS message service (SMS)
SEMI E30	Generic equipment model (GEM)
SEMI E32	Material movement management standard (MMM)
SEMI E32.1	SECS II support for MMM
SEMI E37	High-speed SECS message services (HSMS)
SEMI E37.1	HSMS single services mode (HSMS-SS)
SEMI E37.2	HSMS general services mode (HSMS-GS)
SEMI E38	Cluster tool module communications (CTMC)
SEMI E38.1	Communications environment for CTMC
SEMI E39	Object services standard (OSS)
SEMI E39.1	SECS II support for OSS
SEMI E40	Processing management standard
SEMI E41	Exception management standard
SEMI E42	Recipe management standard (RMS)
SEMI E42.1	SECS II support for RMS
SEMI E53	Event reporting standard

5) EMS

EMS(exception management standard)는 클러스터 콘트롤러의 비정상적인 상태에 대한 reporting과 사용자와의 interaction을 처리하는 예외처리의 개념, 예외처리 상태에서의 장비 운용절차 및 그러한 기능을 제공하기 위한 메시지 서비스 등을 정의한다.

6) ERS

ERS(event reporting standard)는 I/O 레벨 이상에서

의 주기적인 데이터 리포팅과 요구 이벤트 응답에 관한 리포팅의 개념, 운용절차, 메시지 서비스 등을 정의한다.

#### IV. 클러스터 틀 표준화의 발전방향

반도체 산업에서 클러스터 틀의 사용이 급속하게 확산되고 개방형을 추구하는 반도체 장비의 통신 표준화의 추진과 새로운 표준들의 출현 등의 근본적 요인은 IC 제조에 있어 장비의 생산성 향상과 제조 품질에 대한 한계를 극복하려는 공정장비의 통합 및 공장자동화에 대한 강력한 요구가 있기 때문이다.

그러나 이러한 국제표준의 존재에도 불구하고 클러스터 틀의 사용자들은 이들 제품의 상호운용성과 인터페이스 효율에 대해서는 여전히 부정적이며 국제적 표준에 의해 개발된 상용제품들에 대한 정확한 평가를 요구한다. 또한 반도체 공장자동화를 위한 표준으로 GEM이 마련되어 있지만 만족할만한 공장자동화 시스템의 구현을 위해서는 다음과 같이 여전히 해결해야 할 문제가 많이 남아있는 상태다.

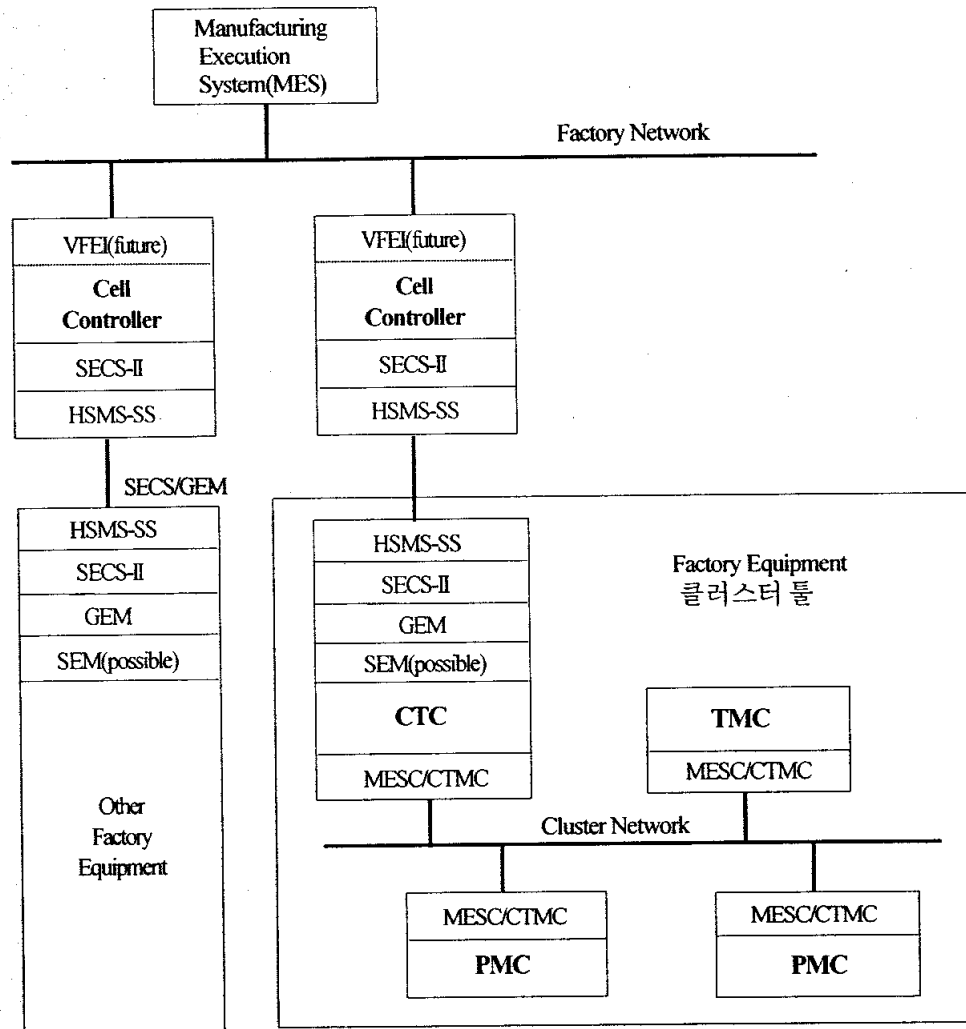
- 시스템 구현의 기술적인 어려움 때문에 개발기간이 길고 비용이 너무 크다.
- 표준이 제공하는 기능이 부족하여 이기종간의 인터페이스가 일치하지 않는다.
- 장비운용의 유일한 표준 시나리오가 제공되지 않아 장비의 통합에 한계가 있다.

위와 같은 문제점을 해결하기 위한 노력은 여러 측면에서 계속되고 있으며 한 예로 SEMATECH에서는 (그림 6)과 같이 미래의 반도체 공장 통신모델을 대비하는 새로운 표준인 VFEI(virtual factory equipment

interface)와 SEM(specific equipment model)의 개발을 진행시키고 있다. VFEI는 반도체 제조공장 운영모델로서 셀컨트롤러(cell controller) 애플리케이션인 공장 호스트 시스템과 반도체장비에 대한 인터페이스를 제공하는 셀컨트롤러 내의 machine drive 사이의 메시지 인터페이스를 정의한다.

또한 SEM은 이기종 장비통합을 위한 별도의 작업을 없애고 미소한 변경만으로 통합이 가능하도록 반도체 장비의 종류와 그것의 운용을 완전하게 정의하는 5개의 표준으로 구성된다. SEM은 반도체 장비모델을 일반화하여 정의하고 있는 GEM 표준을 기반으로 하여 inspection, metrology, stepper, wire bond, molding 등 5가지 클래스의 반도체 장비 운용절차(behavior)와 이들의 프로세스 상태 모델, 정보, 이벤트 및 데이터 리포팅을 상세히 규정하여 GEM 표준을 대체할 것으로 예상된다.

클러스터 틀은 반도체 공장자동화의 작은 축소판이기 때문에 많은 부분에서 공장제어 시스템의 특성과 매우 유사하여 멀지 않은 장래에 이들 두 시스템은 하나의 공장자동화 네트워크로 통합될 것이 예측된다. 클러스터 컨트롤러는 (그림 6)에서 보여주는 바와 같이 공장자동화 네트워크의 한 부분으로 흡수되어 MES(manufacturing execution system)라고 하는 최상위 공장제어 호스트 시스템이 중간제어기인 셀컨트롤러를 통해 직접 클러스터 모듈들을 액세스하여 관리할 수 있게 된다. 이러한 일이 가능하기 위해 클러스터 틀에 있어서도 VFEI, SEM 등의 표준에서 제공하는 기능들을 수용하여 제조공정의 변경에 따라 다양한 프로세스 모듈을 plugin 방식으로 쉽게 접속하여 사용할 수 있으며, 300mm 웨이퍼 등 차세대 디바이스 제조에 필요한 기능의 확장을 제공할 수 있도록 발전하고 있다.



(그림 6) 반도체장비 통합에 의한 공장자동화 표준화 모델

## V. 결론

현재 클러스터 톨의 공급을 주도하고 있는 미국의 장비 제조업체들은 클러스터 컨트롤러의 독자적인 개발보다는 Brooks Automation, Realtime Performance, Texas Instruments 등에서 제공하는 트랜스포트 플랫폼이나 소프트웨어 솔루션에 자신들의 애플리케이션을 개발하여 제공하고 있다. 그러

므로 클러스터 컨트롤러 기술은 이들 극소수의 업체들에 의해 독점되고 있다고 해도 과언이 아니다. 그러나 클러스터 톨 개발의 원래 목적인 이기종 반도체장비의 통합은 여전히 동일한 메이커의 제품 내에서만 가능하다. 이러한 이유는 MESC 표준의 소프트웨어가 장착된 트랜스포트 모듈에 다른 공급자의 클러스터 모듈을 연결할 수 있는 기술적인 가능성은 열어놓고 있으나 제공업체들의 시장점

유희 경쟁과 같은 상업적인 이유가 장애의 주된 요인이다.

클러스터 컨트롤러는 반도체 장비기술의 핵심이라 할 수 있으나 국내에서는 클러스터 장비보다는 단위공정장비의 국산화 문제가 해결해야 할 시급한 당면과제로 직면하고 있으며, 더욱이 클러스터 툴의 국내수요가 아직까지 크게 형성되어 있지 않아 국내개발은 활성화되지 못하고 있다. 최근에 한국전자통신연구원과 일부 중소기업에서 클러스터 툴의 개발을 시도하고 있으나 아직 상품화에는 못미치고 있다. 디바이스 제조에 있어서 클러스터 툴의 경제적 이득과 클러스터 툴이 반도체 제조공장의 단위공정 자동화의 표준 모델로 자리잡고 있음을 감안하면 국내에서도 이제 클러스터 툴의 개발과 함께 관련된 표준화 활동에 본격적으로 참여하여 클러스터 툴의 선진 기술을 조기에 흡수하고 시급히 국내시장의 기반을 마련하여야 할 것으로 본다.

[6] Pieter Burggraaf, "The Status of Equipment Communications," *Semiconductor International*, December 1995, pp. 56-62.

## 참고문헌

- [1] Equipment Automation/Software Volume I, II, Book of SEMI Standard, 1996.
- [2] Kevin Nguyen, "The Development and Implementation of a Cell Controller Framework," *IEEE/SEMI Int'l Semiconductor Manufacturing Science Symposium*, 1993, pp. 54-57.
- [3] Pete Singer, "Cluster Tool Communications: MESC is Back," *Semiconductor International*, June 1994, pp. 94-100.
- [4] Katherine Derbyshire, "Applications of integrated processing," *Solid State Technology*, December 1994, pp. 45-49.
- [5] Jong Hyun Lee, "Development of a MESC-Compliant Cluster Tool," *Solid State Technology*, October 1995, pp. 93-97.