

# PCB 공정에서 데이터 변환 인터페이스 구축

## Data Transformation Interface Construction in PCB Product

이승혁, 한정수, 한군희  
천안 대학교

Lee Seung-Hyuk, Han Kun-Heui, Han Jung-Soo  
Division of Information & Communication,  
Cheonan Univ.

### 요약

본 연구는 PCB 생산 자동화를 위한 데이터 변환 시스템을 구현하였다. CAD로 설계한 데이터는 PCB 생산라인과의 데이터 호환을 이루지 않기 때문에 CAD로 설계한 PCB 부품 정보를 분석하여 Human Error 검출 알고리즘을 개발하였고, 자동화를 위한 IC 부품들의 정보를 DB로 구축하였으며 PCB 생산 라인에 적합한 데이터로 변환하는 알고리즘과 부품의 정보 추가, 수정하기 위한 시스템을 구축하였고, 사용자 인터페이스를 설계 및 구현하였다. 기존의 수작업 공정을 자동화함으로써 1-2일 정도의 시간을 수분 이내로 단축하였고 데이터의 신뢰성과 효율적인 PCB 생산 라인을 향상시킬 수 있었다.

### Abstract

In this paper, we designed data transformation interface for the automation of PCB product. The data designed to CAD does not exchange itself for the assembly line, so we analyzed the information of PCB components designed to CAD and developed human error detect algorithm. We also constructed database for the information of IC components. Developed the algorithm can change to the data suitable for the assembly line of PCB and this system was designed to do addition and revision of the components through the user interface. By automating existing manual processing, we were able to shorten the time of 1-2 days to several minutes and enhance reliability of the data and the efficient assembly line of PCB

## I. 서론

본 논문에서는 현재의 PCB 생산에 있어 데이터 변환을 위한 수작업을 자동화하고 여러 가지 에러 검출을 함으로 비생산적인 현지 중소기업의 고통을 해결하고자 PCB 생산 자동화를 위한 데이터 변환 시스템을 구현하였다. 현재 PCB 생산에서는 하청업체로부터 받은 PCB 설계 데이터는 CAD로 만들고 있다. 따라서 이렇게 설계된 CAD데이터는 PCB 생산라인과의 데이터 호환이 이루어지고 있지 않는 실정이다.

즉, PCB 생산을 위해 Loader, Surface Mount Clean Machine, Screen Printer, Solder Print Tester, Chip Mounter, Tray Feeder 등의 설비 장비를 통하여 생산된다[1]. 그러나 CAD로 설계한 데이터는 수작업에 의해 수행하다보니 Human Error가 발생하고 이로 인하여 정확도가 떨어져 불량 제품이 생산되는 경우가 많아 어려움을 겪고 있다. 수작업 과정은 하나의 PCB 생산을 위한 데이터 변환만 약 8시간이 걸리고, 테스트 공정까지 포함하면 1-2일 정도가 걸린다.

따라서 본 연구에서는 PCB 조립 공정에서 CAD로 설계된 IC 부품별 데이터를 분석을 통한 설비와의 데이터 호환을 위한 시스템을 설계, 구축하여 자동화도 작업능률을 극대화하기 위한 것이 그 목적이다.

## II. 데이터 변환 시스템

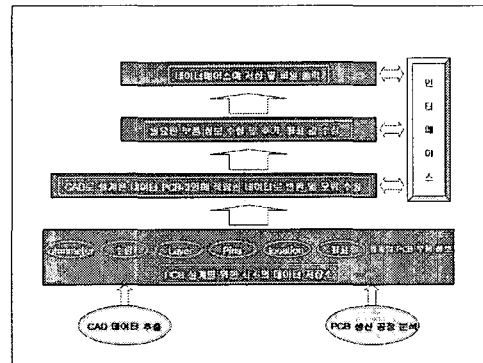
### 1. 시스템 구조

CAD로 설계된 데이터들은 보드의 앞면 혹은 뒷면 중 어떤 위치에 결합될 것인지에 대한 데이터들을 가지고 있으며, 보드에 총 부품의 개수나 그 부품의 특징들에 대한 데이터를 가지고 있다. 하지만 이러한 데이터들을 PCB 생산 라인과 전혀 호환을 이루지 않다. 왜냐하면 CAD 설계자는 PCB 생산 라인을 이해하지 못한 상태에서 자신만의 설계 방법에 따라 설계하기 때문이다. 이는 똑같은 PCB를 설계할 때에도 설계자에 따라 다른 결론의 CAD 데이터가 나타날 수 있다는 의미가 된다. 따라서 이러한 상의한 데이터를 PCB 라인에 적합한 데이터로 변환하는 과정이 필요하다. 따라서 PCB 제조업체는 이러한 정보를 다시 수작업으로 하나하나 모두 변환하여 PCB 라인에 적합한 데이터로 작성해야하는 부담을 안고 있다. 이러한 비효율적인 작업은 데이터를 변환하는 PCB 산업체의 고비용과 생산성 감소에 직접적인 연관을 가지고 있다. 뿐만 아니라 수작업을 통해 발생할 수 있는 Human Error는 엄청난 손실을 초래할 수 있다.

따라서 그림 1에서처럼 CAD 정보 분석을 통해 부품 정보를 추출한다. 또한 PCB 생산 라인의 전 과정을 분석하여 필요한 PCB 부품에 대한 정보를 획득하며 부품별로 데이터를 분류한다[2]. 저장소에 구축될 정보 안에는 품목 명, 품목번호, 위치, 각도, 규격, 수량 등의 데이터들이 포함된다. 저장소는 인터페이스를 통한 부품 검색을 위해 키워드를 정의하고 데이터 변환 프로그램을 위한 준비를 완료한다. 이 과정에서 CAD 데이터의 오류 수정을 하게 되는데 CAD 설계자에 의해 설계된 데이터가 설계자의 Human Error

에 의해 잘못된 데이터를 가질 수 있기 때문이다.

이런 문제를 해결하기 위해 Human Error 검출 알고리즘을 이용하여 저장 데이터를 검증 받는다. 검증 과정에서 Error가 발생하면 이를 인터페이스 사용자에게 알리고 CAD 데이터 중에서 Error가 발생한 데이터를 다시 입력받게 된다. 이런 과정을 통해 Error가 없는 데이터에 대하여 필요한 부품의 정보를 추가하게 된다. 이 부품 정보는 이미 PCB 생산 공정을 분석하여 얻은 데이터로부터 가지고 오게 된다. 또한 CAD 설계자는 PCB 보드의 좌표를 자신의 방법에 따라 정하는 반면, PCB 생산 라인은 보드의 한쪽 모서리를 원점으로 갖고 좌표를 계산한다. 따라서 CAD 데이터의 좌표 값들을 PCB 생산 라인에 적합한 좌표 값들로 변환하게 된다. 좌표 값 변환은 인터페이스를 통해 사용자에게 현재의 CAD 데이터의 좌표 값이 어디를 원점으로 정하여 설계되었는지를 알려주고 사용자는 PCB 생산 라인의 원점을 확인하는 방법으로 원점을 수정하게 된다.

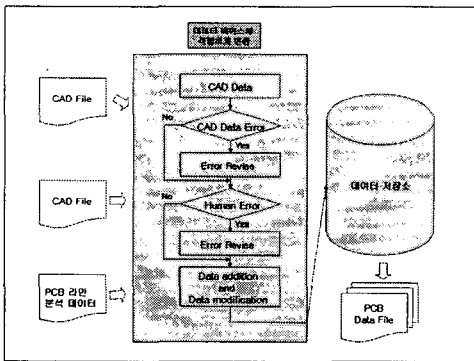


▶▶ 그림 1. 데이터베이스 변환 시스템 구조

### 2. CAD와 PCB 데이터 추출

CAD 데이터는 CAD 설계자의 편의에 의해 PCB 생산 라인과는 관계없는 불필요한 데이터를 가지고 있다. 이런 불필요한 데이터는 PCB 생산 라인의 공정 분석을 통해 얻을 수 있으며 분석된 결과에 따라 CAD 데이터 중 실제적으로 필요한 데이터만을 추출

하게 된다. 뿐만 아니라 CAD 데이터 중에서 설계자의 편의에 의해 삭제된 일부 데이터들이 PCB 생산 라인에서는 유용한 정보일 수 있다. 따라서 이렇게 추가, 삭제된 데이터에서 불필요한 데이터는 삭제하고 필요한 데이터는 다시 추가하여 추출하게 된다[3]. 예를 들어 CAD 설계자는 Vendor에 대한 정보를 상세히 기록하여 CAD 데이터에 삽입할 수 있다. 하지만 Vendor는 PCB 생산에 어떤 영향도 미치지 않기 때문에 불필요한 Vendor에 대한 정보는 삭제하게 된다. 또한 'Specification'란 데이터가 없을 경우 이는 앞에서 이미 CAD 설계자가 'Specification'에 대한 데이터를 정의하였기 때문이다. 하지만 데이터 저장소에 그대로 저장하게 된다면 이는 NULL로 인식하게 된다. 따라서 앞에 이미 정의된 데이터를 찾아내어 삽입하게 된다. 이처럼 예를 들어 설명하였지만 이러한 필요 혹은 불필요한 데이터는 수없이 많이 있다. 이를 하나하나 찾아내어 수정하여 PCB 데이터 저장소에 저장하게 된다. 그림2는 CAD 데이터를 추출하여 데이터 저장소에 저장하여 CAD 설계자에 의한 CAD 데이터 오류나 Human Error를 제거하는 단계를 보여주고 있다. 이 단계에서는 저장소에 오류 없는 데이터를 저장하기 위함이다.

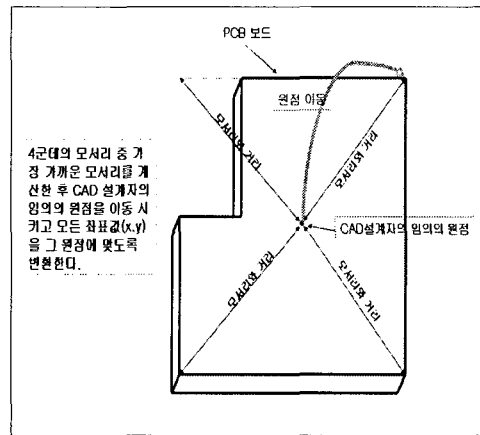


▶▶ 그림 2. 저장소에 적합한 데이터로의 변환

### 3. 좌표 수정과 Layer 분리

하나의 Table로 저장된 데이터에서 PCB 부품별

데이터는 CAD와 같은 데이터들을 포함하고 있지만 CAD 데이터의 좌표 값(x,y)과 PCB 생산 라인에서의 좌표 값(x,y)는 서로 다른 체계로 되어있기 때문에 각각의 좌표 값이 어떤 위치를 중심으로 설계되었는지 확인할 필요가 있다. CAD 설계자는 좌표 값을 PCB 보드의 중간정도의 위치를 원점으로 정하는데 반면에 PCB 생산라인에서는 PCB 보드의 모서리 부분을 원점으로 사용하고 있다. 따라서 CAD 설계자가 어떤 위치를 원점으로 정했는지 원점 알고리즘을 통해 PCB 보드의 네 군데의 모서리 중에서 가장 가까운 모서리를 찾아 그 거리를 계산한 후 인터페이스를 통해 사용자에게 거리에 대한 오차를 입력받도록 요구하고 이 오차에 대하여 원점 알고리즘은 모든 좌표 값을 수정하게 된다. 그림 3은 CAD설계자의 임의의 원점을 계산하여 이동하는 모습을 보여주고 있다.



▶▶ 그림 3. 원점 수정

좌표 값이 수정된 데이터에서 Layer 값에 따라 새로운 다른 Table에 저장하게 된다. Layer의 값은 PCB보드의 앞면 혹은 뒷면에 IC부품을 조립할 것인 지에 대한 값을 가지고 있기 때문이다. Layer의 값이 Top의 경우 앞면에 조립되고 Bottom의 경우 뒷면에 조립되기 때문에 각각을 분리하여 저장하게 되며, 누락이 된 데이터에 대하여 Layer의 값을 입력받도록

사용자에게 알려 새로운 Layer의 값을 입력받는다. 이는 PCB 생산라인이 Top과 Bottom으로 구분되어 있기 때문이다. 또한 생산라인은 하나의 텍스트 파일로 데이터를 입력하게 되는데 그 또한 Layer 값에 따라 하나 파일로 입력받도록 되어있다.

#### 4. 데이터베이스 및 파일 출력

지금까지 여러 개의 데이터베이스 내의 Table에 필요에 따라 저장된 정보들을 통합하고 텍스트 파일로 출력하는 작업을 하게 된다[4]. 데이터베이스는 사용자의 편의성을 고려하여 흔히 사용하는 Access로 설정하였다. 이는 사용자가 데이터 변환에 있어 그 과정을 쉽게 알아볼 수 있으며 Excel 파일로의 변환도 자유로워 각각의 부품에 대한 정보를 사용자가 쉽게 찾아 볼 수 있기 때문이다. 최종 PCB 생산 라인에 필요한 부품 데이터만을 읽어 최종 텍스트 파일로 출력한다. 텍스트 파일 출력이 있어 한 라인에 하나의 부품 정보만을 출력한다. 부품 정보의 출력 데이터는 Location No, Specification, Orient, X, Y, Layer 순이며 각각의 값은 Tab 간격으로 구분하였다.

#### 5. 인터페이스

인터페이스 설계에 있어 데이터베이스에 저장된 데이터들을 사용자가 쉽게 접근하여 데이터의 검증이나 확인을 받을 필요가 있다. 이는 Human Error를 방지할 수 있을 뿐만 아니라 수정이 필요한 데이터에 대해서도 수정을 쉽게 할 수 있도록 하기 위함이다. 따라서 본 연구에서는 사용자에게 쉽게 검증 및 확인을 할 수 있고 시스템에 의해 변환된 과정을 쉽게 알아볼 수 있도록 인터페이스를 설계하였다. 이 인터페이스는 네트워크나 인터넷이 특별히 필요 없는 관계로 PCB 제작자의 편의성을 고려하여 운영체제는 윈도우를 이용하였으며, 데이터베이스에 접근할 수 있는 인터페이스의 개발을 위해 데이터베이스는 Access를 이용하여 OLE-DB로 데이터베이스 응용 프로그램을 만들어 접근하였다. 여기서 Access를 이

용한 것은 사용자가 쉽게 사용할 수 있다는 장점도 포함된 결과이다. OLE-DB는 COM 서버의 기능을 이용하여 COM 클라이언트 프로그램을 만들 것이므로 MFC보다는 ATL을 이용하는 것이 효율적이지만 ATL이 사용자 인터페이스에 관련된 기능을 거의 지원하지 않는 관계로 MFC와 ATL(Active Template Library)을 모두 이용하였다.[5] 그림4는 본 연구에서 구축한 데이터변환 시스템의 결과이다.

Join	Specification	Orient	X	Y	Layer
C86	16R.V5V.0.HF.2.25V	0.0	46.6	101.0	Top
C85	16R.V5V.0.HF.2.25V	180.0	73.3	83.0	Top
C84	16R.V5V.0.HF.2.25V	180.0	64.4	133.0	Top
C83	16R.V5V.0.HF.2.25V	180.0	64.4	134.0	Top
C82	16R.V5V.0.HF.2.25V	90.0	74.3	143.0	Top
C81	16R.V5V.0.HF.2.25V	90.0	87.8	142.0	Top
C80	16R.V5V.0.HF.2.25V	90.0	95.4	142.0	Top
C79	16R.V5V.0.HF.2.25V	0.0	116.0	126.0	Top
C78	16R.V5V.0.HF.2.25V	90.0	89.2	142.0	Top
C77	16R.V5V.0.HF.2.25V	180.0	64.4	124.0	Top
C76	16R.V5V.0.HF.2.25V	90.0	85.0	143.0	Top
C75	16R.V5V.0.HF.2.25V	90.0	94.0	147.0	Top

▶▶ 그림 4. 데이터 변환 결과

### III. 결론

본 연구의 목적은 사용자가 쉽고 편리하게 CAD 데이터를 PCB 생산 라인에 적합한 데이터로 변환하는 것이다. 그동안 이런 과정은 수작업에 의해 1-2일이 소요되었으며 직접 눈으로 테스트하는 과정을 거쳐야 했기 때문에 많은 시간과 불량 생산할 가능성을 갖고 있었다. 하지만 본 연구에서 이러한 과정을 획기적으로 보완하였고 정확한 PCB 생산을 위해 여러 가지 검증 알고리즘과 신뢰성을 확보할 수 있도록 PCB 생산 전 과정을 분석하는 한편 CAD 데이터 내에 Human Error를 검출하고 PCB 생산 전 과정을 분석한 데이터를 비교하여 최소한의 오류까지 제거하였다. 이렇게 오류를 제거한 데이터들을 변환 알고

리즘을 통해 CAD 데이터를 PCB 생산라인에 사용하는 데이터로 변환할 수 있도록 하였고, 이를 사용자가 쉽게 접근하고 변환 과정을 확인할 수 있는 인터페이스를 설계 및 구현하였다. 이로서 정확하고 효율적인 PCB 생산을 위한 이기종간의 데이터 호환을 통한 공장 자동화를 완성하였다

### ■ 참고 문헌 ■

- [1] <http://www.qmaxtest.com>
- [2] 송영재, “객체지향모델링과 CBD중심 소프트웨어공학”, 이한출판사, 2004.
- [3] C.Batini, s. Ceri, and S.B. Navathe, “Conceptual Database Design : An Entity-Relationship Approach”, Benjamin Cummings, 1992.
- [4] K.EI Emam and N.H. Madhavji “A Field Study of Requirements Engineering Practives in Information Systems Development”, York, England, 1995.
- [5] 박광우, “Visual C++.Net Programming Bible”, 삼양미디어, 2004.