

# 경계표현법을 응용한 순차 금형 자동 설계시스템 개발 및 적용

김용연<sup>#</sup>

## A Development of the Design System of the Progressive Stamping Dies by using Boundary Representation and Its Application

Yong Yun Kim<sup>#</sup>

### ABSTRACT

In this paper, an auto-design system is introduced for a stamping tool based on commercial computer aided design system with its drafting language. The auto-design system consists of tool oriented product design subsystem which modifies and configures the product drawing, tool concept design subsystem which make a design of the punches and their punching progression and parts design subsystem that makes automatic dimension. The system is applied to the mechanical design of the stamping tool. The main logic of the system is based on half-edge theory, a kernel for the 3 dimensional CAD system, which is applied to 2 dimensional drafting auto-design system. The auto-design system enables to conspicuously reducing the designing time of the tool. In addition, there is little drafting error that had been about 3% without auto design program. It is effective to reduce the development time for new products because of rapid designing time of the tools, standardization of the stamping tool and the drafting rule for the auto-design system. The auto-design system yields high efficiency of the tool manufacturing system.

**Key Words** : Auto-Design system (자동설계시스템), Lead Frame (리그프레임), Stamping tool (타발금형), Punching order (타발순서)

### 1. 서론

전단공정으로 생산되는 부품의 품질과 생산성은 생산금형에 영향을 많이 받고, 신규 금형을 개발할 때 가장 큰 비중을 차지하는 부분은 금형설계이다. 일반적으로 금형설계를 하기 위하여 범용 CAD S/W를 이용하고 있으나, 범용 CAD S/W에 의한 수동적 설계는 자동설계에 비하여 설계하는 데

시간이 많이 소요될 뿐 아니라 설계오류도 자주 발생시킨다. 이러한 문제를 해결하기 위하여 범용 CAD 시스템을 이용한 전용설계 프로그램이 최근 많이 개발되고 있다.1,2 생산성을 향상시키기 위해서는 설계공정을 표준화하는 것이 필수적이지만 설계자의 경험과 직관에 의존한 수동 설계 시스템에서의 교육을 통한 설계공정 표준화는 한계에 이르게 된다. 즉, 동일한 기계일지라도 설계도면은 설계

접수일: 2004년 7월 27일; 게재승인일: 2004년 11월 19일

# 교신저자: 충북대학교 기계공학부

E-mail [yykim@cbnu.ac.kr](mailto:yykim@cbnu.ac.kr) Tel. (043) 261-3353

자의 직관과 경험에 따라 서로 다르게 표현될 수 있고, 설계자가 미적추구를 위한 비공학적인 부분도 있다. 이러한 문제는 자동설계프로그램을 개발하여 적용시키는 데도 가장 어려운 부분이다. 이러한 표현의 다양성은 프로그래머와 설계자의 많은 토론을 통하여 설계표준을 정립함으로써 단순화할 수 있다. 설계표준은 제품도로부터 금형상세도에 이르기까지 일련의 설계 공정을 자동화 프로그램 개발을 가능하게 하였다.

본 논문에서는 Fig. 1과 같은 반도체 리드프레임 생산금형의 설계자동화 프로그램을 개발하여 적용시킨 결과를 기술하였다. 자동설계를 위해서는 단순 제도의 자동화를 포함하여 금형구조해석<sup>3</sup>, 편치좌굴해석<sup>4</sup>, 등이 동시에 수반되어야 하나 본 논문에서는 금형제도 자동화에 대해 주로 기술하였다.

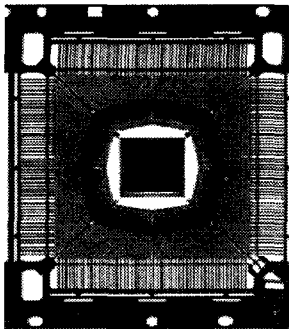


Fig. 1 208pin IC Lead Frame

## 2. 자동설계 시스템 환경 및 요소기술

### 2.1 자동설계 시스템 환경

금형 자동설계 프로그램은 범용 CAD s/w인 I-Deas Drafting 모듈, CAD 지원 프로그램(I-Deas language)<sup>6</sup>과 C 언어로 코딩한 복합모듈로 구성되어 있다. CAD 지원 프로그램은 사용자가 새로운 기능을 추가할 수 있는 CAD 자체 지원프로그램이고, 새로운 도형을 제도하거나, 제도된 도형에 대한 정보를 추출하기 위해서 사용된다. 지원프로그램은 C 언어로 코딩된 외부 서브프로그램을 실행시킬 수 있다. 외부 서브프로그램은 CAD 자체 지원 프로그램으로 처리하기 어려운 복잡한 로직을 효과적으로 수행하기 위해 사용하였고, 처리속도에 있어서 자체 지원 프로그램에 비해 효과적이고, 로직

구현에 있어서도 용이하였다. 프로세스간의 통신은 Fig. 2와 같이 주 프로그램과 서브프로그램의 관계로 이루어져 데이터 파일을 이용하여 통신하는 방법을 채택하였다. 주 프로그램이 외부서브프로그램을 실행하여 서브프로그램 종료 후, 그 결과와 데이터를 이용하여 I-Deas 프로세스를 진행하게 된다. 그러므로 주 프로그램이 그 내부에서 모듈화된 서브프로그램을 실행시킨다. Fig. 2는 실행 프로세스를 도식화한 것으로 주 프로그램과 C 프로그램이 데이터 교환을 하기 위해서는 두 프로그램은 규정된 파일 형식이 필요하다. 파일형식이란 일반적 도형요소(entity)에 대한 위상정보와 형상인식 프로그램에 의해 인식된 도형을 표현하는 파일구조이다. 도형요소란 도형을 구성하는 선, 원, 호등의 단위 개체를 의미한다. I-Deas로부터 기하학적 정보 추출하여, 외부프로그램을 탑재할 수 있는 형식으로 데이터베이스를 개발하였다.

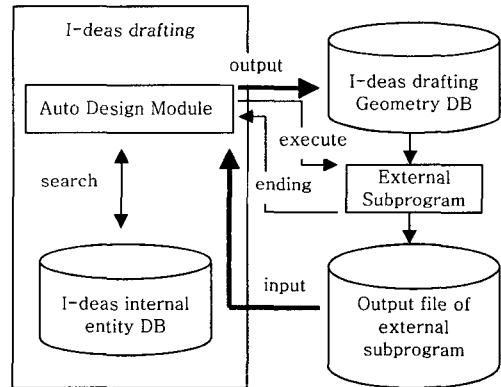


Fig. 2 Diagram of auto design module

### 2.2 형상인식 프로그램

순차이송형 금형에 있어서, 펀치 및 다이에 관련된 많은 부품형상은 CAD상에서 페루프로 연결된 일련의 도형요소의 집합으로 표현된다. 2차원의 CAD는 형상을 인식하기 위한 위상정보를 지원하지 않는다. 따라서 I-Deas로부터 도형정보를 추출하고, 경계표현법(B-rep)을 응용하여 형상인식 프로그램을 개발하였다. 경계표현법은 3차원 솔리드 모델링에서 사용되는 이론으로 입체의 내부와 외부를 경계짓는 면, 모서리, 절점의 집합으로 Fig.3 ㉔와 같은 입체를 표현하는 방법이다. 입체도형은 Fig.3 ㉕와 같은 부피가 0인 3차원 쉬트(sheet) 모델로 나타낼 수 있는 데, 이러한 쉬트모델은 2차원 평면상

의 도표로 표현하기에 적합하다. 이러한 도표 자료 구조를 구현하기 위해 경계표현법 자료구조의 하나인 Half-edge 자료구조를 채용하여 2차원에서 입력 요소의 위상관계를 표현하였다. Fig. 3⑥의 쉬트모델에서 각 면은 그 구성모서리를 반시계방향으로 순회하는 벡터의 집합으로 정의하면, 각 모서리는 서로 반대 방향의 벡터가 쌍으로 생성되는데, 이러한 가상적인 벡터가 half-edge이다. half-edge로부터면, 모서리, 정점의 위상요소를 표현할 수 있다. 이러한 개념을 2차원 CAD에 적용함으로써 얻을 수 있는 새로운 기능을 도입함으로써 첫째로 2차원 CAD에서 제공되지 않은 도형요소간의 위상관계를 일관되게 표현할 수 있다. 페루프와 열린루프의 인식이 동일한 자료구조로 저장 및 처리할 수 있다. 금형부품의 형상은 대부분 페루프로 구성되나, 설계 오류로 인하여 형상이 열린루프로 된 경우가 있다. 열린루프는 자동설계공정에서 오류를 발생시키기 때문에 검색을 하여 페루프로 수정되도록 하였다. 두 번째로 도형의 위상정보를 조합하여 특정 형상에 대해 정의함으로써 조립도면으로부터 형상을 자동 추출할 수 있다. Fig. 4④와 같은 분할형 다이의 경우 다이인서트(die insert)를 독립적으로 인식할 수 있을 뿐 아니라, ⑤와 같은 조립체에서 다이인서트를 개별적으로 추출할 수 있다. Half-edge 자료구조로서 형상을 구성하는 모든 도형요소들 서로 반대인 2개의 방향성을 부여하고, 도형요소를 방향성에 따라 2개로 분할하여 독립적으로 위상정보를 저장한다. 또한, 절점으로 연결되어 있는 모든 도형요소들의 위상정보는 하나의 집합체로 인식될 수 있도록 데이터베이스 내에서 구분된다.

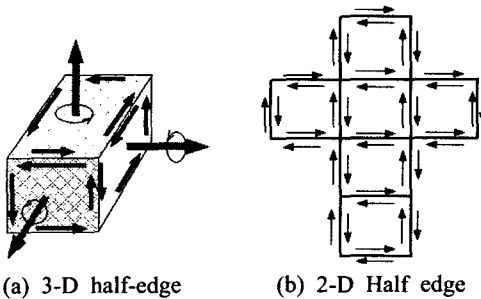


Fig. 3 Example of Half edge

Half-edge 자료 구조는 2차원 CAD 상에서 도형요소들의 집합을 페루프화하여 부품이라는 개체를 인식시킬 수 있는 수단을 제공한다. Fig.3에서 3차원

직육면체에 정의된 half-edge는 직육면체를 2차원 평면상에 전개하였을 경우에도 동일하게 정의될 수 있으며, 2차원 전개도에서 half-edge 자료 구조로부터 도형을 구성하는 각 면을 개체로 추출할 수 있다. 따라서 Fig. 4의 die insert를 인식시킬 때 half-edge 구조를 이용하면 ④와 같은 독립형상이 인식될 수 있을 뿐만 아니라 ⑤와 같은 조립체에 대해서도 부품①과 부품②의 형상으로 개별 추출될 수 있다.

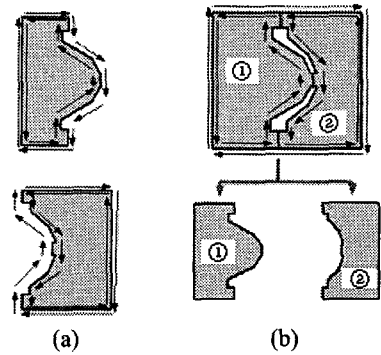


Fig. 4 Die insert

### 2.3 공간인식 프로그램

편치를 설계하기 위해서는 리드와 리드사이의 공간을 인식할 수 있는 기능이 필요하다. 공간인식을 하기 위해서 (1) 내부 리드를 구성하는 요소에 일정한 방향성을 부여하기 위해 페루프 상태로 정렬시켜 반시계 방향 half-edge 데이터를 생성시키고, (2) Half-edge 데이터에 의해 방향성을 갖는 리드 선단 직선들의 배열상태를 정의한다. 리드 선단 직선들의 시작점과 (0,0)점을 연결하는 가상선의 각도를 산출하여 각도의 크기 순으로 리드 선단 직선들 상에 순번을 지정하여, (3) 연이은 두 선단 직선 사이를 공간으로 인식하고, 그 공간을 형성하는 리드의 입력인자들을 탐색한다.

### 3. 자동설계 시스템 적용 및 고찰

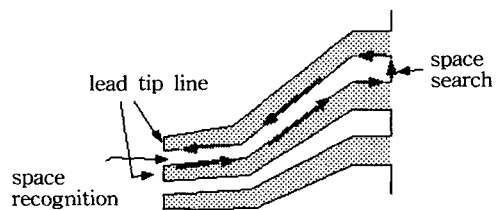


Fig. 5 Space recognition model

### 3.1 순차형 금형설계 개요

순차이송형 금형은 Fig. 6처럼 펀치조립부, 스트리퍼조립부, 다이조립부로 나눌 수 있다. 부록에 표현된 것처럼 금형설계는 일반적으로 개념설계, 조립도, 부품도, 부품가공공정설계의 순으로 진행된다. 제품도로부터 금형을 자동설계하기 위해서는 먼저 제품도가 연속적인 폐곡선으로 구성되어야 한다. 그러나 제품도는 불연속 곡선으로 구성된 경우가 많기 때문에 개념설계에서 먼저 불연속점을 검색하여 연속선으로 수정하고 공간인식프로그램으로부터 공간을 인식하여 펀치인선을 자동설계한 후 블랭킹의 순서가 결정되면 개념설계가 완료된다. 개념설계로부터 라이브러리를 적용하여 펀치조립도, 스트리퍼 조립도, 다이조립도를 생성한다. 마지

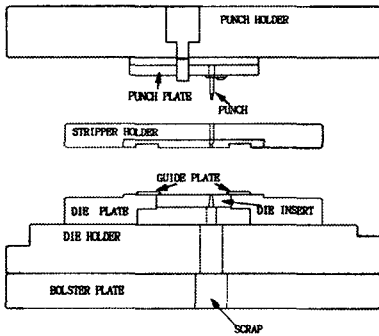


Fig.6 Diagram of progressive dies

막으로 부품도를 생성하고, 각각의 도면에 대한 치수를 자동 기입한다. 또한 CAM가공을 위하여 NC code를 생성한다. 자동설계 프로그램은 부록에 기술된 표준설계 공정을 실행할 수 있도록 각각의 실행 아이콘을 갖고 있고, 프로그램의 실행 후 설계변경을 용이하도록 하기위해 독립적으로 수행할 수 있도록 하였다. 제품 형상이 규칙적이지 않은 부분은 자동화하기 어렵기 때문에 표준화할 수 없는 부분에 대해서 수동으로 설계할 수 있도록 하였다.

### 3.2 공간인식 및 펀치인선 설계

제품도에서 공간은 펀치의 형상을 의미한다. 따라서 리드프레임 제품도로부터 공간을 자동인식한 후 펀치인선을 생성하는 과정을 Fig. 7에서 나타내고 있다. Fig. 7(a)와 같이 2개의 리드를 형성하기 위한 공정이 Fig. 7(b)~(d)이다. 리드와 리드사이의 공간을 펀치인선으로 생성하고, (c)와 같이 리드보

다 길게 설계한다. 펀치1과 펀치2를 타발 후 소재에 발생하는 잔류응력이 리드의 끝단부에서 가장 크기 때문에 타발 후 리드의 변형을 최소화하기 위함이다. Fig. 8에서처럼 펀치끝단을 연장하기 위한 리드의 선단으로부터 연장되는 펀치 직선 길이(D1)과 리드폭이 늘어나는 양을 고려한 펀치폭 확장(D2)과 펀치끝단부의 모서리 가공반경(D3)이 설계인자이다. 각 인자 기본값은 표준화되어 있으나, 리드와 패드사이의 거리에 따라 설계자가 결정해야 하는 경우가 많다. 펀치설계인자를 결정하여 입력하면, 리드를 형성하는 페루프가 Half-edge형식으로 인식하여, 리드와 리드사이의 공간을 인식하고, 내장된 설계규칙에 따라 펀치를 설계한다. 리드끝 부분의 설계에 있어서 사용자가 설정한 반경이 부적합하면 프로그램에 의해 자동 수정된다. Fig. 9는 리드프레임 설계도로부터 내부리드 펀치 설계를 나타낸 것으로 수동으로 설계할 경우 1시간 정도 소요되었던 작업이 자동설계프로그램을 적용할 경우 30초로 단축되었다. 리드가 많을수록 프로그램의 효과는 더 크다. 펀치인선부 형상이 생성되고 블랭킹 순서를 사용자가 결정하면, 순서에 따라 펀치를 전개시키고, 금형 전체 길이를 결정한다. 경험적인 데이터와 수치해석을 필요한 공정은 기술 데이터를 데이터베이스화 하는 것이 효율적이다.<sup>1,2</sup> 따라서 펀치와 다이인서트의 강도 해석, 진동해석은 자동설계프로그램에서 수행하지 않고 해석결과를 규칙

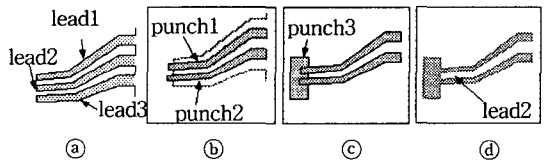


Fig. 7 Stamping process of internal lead

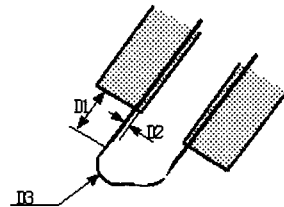


Fig. 8 Design parameter of punch

화하여 데이터베이스화 하였다. Fig. 10은 제품도에 대한 펀칭 순서도로 표준제품에 대해서는 자동

으로 설계하도록 하였고 표준제품이 아닌 경우는 수동으로 설계자의 경험을 반영할 수 있도록 하였다.

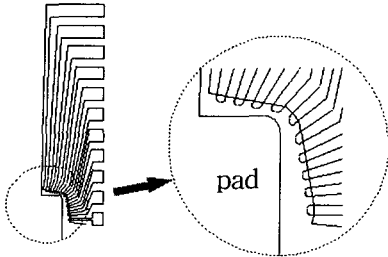


Fig. 9 Punch design

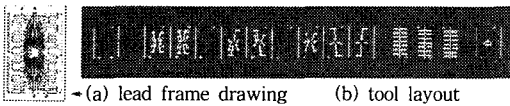


Fig. 10 Tool layout

### 3.3 조립도 설계

조립도 자동설계 프로그램에서 표준금형부품의 데이터베이스 구축을 위해 형상과 크기가 변하지 않고 고정된 부품에 대해서는 심볼(기호)을 사용하였고, 도형의 형상과 크기를 변화시킬 필요가 있는 부품에 대해서는 크기와 형상을 일률적으로 변화시킬 수 있도록 가변형 방법을 적용하였다. 고정기호는 하나의 도형요소처럼 사용되며, holder와 plate 부품의 설계와 체결부품에 적용시켰다. 금형 레이아웃의 끝부분을 지정하면 홀더 심볼이 자동 검색되어 삽입된다. 가변형 도형은 설계 데이터 베이스에서 부품의 형상을 미리 정의하고 특정 치수를 설계인자로 제어할 경우, 가변적인 설계 조건에 부합되는 형상설계가 반복적으로 수행된다. Fig.11의 Guide plate 형상에 대한 가변 프로그램은 Fig.12와 같다. Guide plate를 구성하는 요소의 위상정보는 모두 함수로 정의하고, 설계자가 입력하는 정보에 의해 수행된다. Guide plate는 한 금형에서 여러 개로 구성되어 각각의 길이가 다르고, 소재의 위치를 제어하는 파일럿의 수도 다를 수 있기 때문에 오류가 자주 발한 부분이다. 프로그램에 의하여 자동설계됨으로서 설계오류는 거의 발생되지 않았을 뿐 아니라 설계시간도 크게 단축되었다.

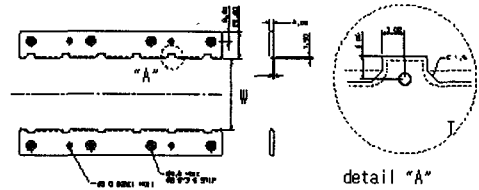


Fig.11 Guide plate design

```
# create line (-FD-1.5), MatWtH0, H20.5, >P(1)-1.5, MatWtH0, H20.5, entU(1)
# create line (-FD, MatWtH(0, H20.5)-1.5, -FD, MatWtH0, 1, entU2(1))
# create line (-FD-1.5), MatWtH0, 1-1.5, >P(1)-1.5, MatWtH0, 1-1.5, entUS(1))
# create line (>P(1), MatWtH(0, H20.5)-1.5, >P(1), MatWtH0, 1, entU4(1))
# create line (-FD, MatWtH(0, H20.5)-1.5, -FD-1.5), MatWtH(0, H20.5), entUS(1))
# create line (-FD-1.5), MatWtH0, 1-1.5, -FD, MatWtH0, 1, entUB(1))
# create line (>P(1)-1.5, MatWtH0, 1-1.5, >P(1), MatWtH0, 1, entU7(1))
# create line (>P(1)-1.5, MatWtH(0, H20.5), >P(1),
MatWtH(0, H20.5)-1.5, entUS(1))
```

Fig.12 Guide plate geometry coding

### 3.4 부품도 설계

부품도 설계 일괄처리 프로그램은 설계자가 작업을 하고 있는 디렉토리에 있는 파일중 이름에 『\_dtl.prg』의 문자열을 갖고 있는 파일이 있으면 작업을 시작된다. 부품의 종류를 자동으로 인식하며, 도면의 크기가 선정되고, 도면의 레이아웃 구성되고, 치수기입, 등이 수행된다. 부품도 데이터 리스트에 따라 각 부품들이 어떤 부품인가를 인식하고, 그 부품에 맞는 부품도 설계 서브프로그램을 실행시키는 일만 한다. 예를 들어 리스트 중에서 P01\_dtl.prg라는 항목을 읽게 되면, 이 데이터 파일이 펀치라는 품명을 갖는 부품이라는 것을 인식하고, 펀치 부품도를 설계할 수 있도록 탑재되어 있는 서브프로그램에 P01\_dtl.prg라는 데이터를 보내 주고, 실행하도록 명령을 내리는 것이다. 구동된 서브프로그램에 의해 한 부품에 대한 설계가 완료되면 위와 같은 작업은 모든 부품이 설계될 때까지 반복 수행된다.

### 3.5 상세도 설계

상세도 필요에 대한 판단기준으로 도면을 육안으로 식별할 수 있는 가장 작은 선의 길이를 결정하고, 상세도에서 표현될 가장 작은 선의 길이를 비교 판단하여 상세도가 필요한 경우 영역을 확보하여 상세도를 작도한다. 따라서 일차적으로 상세도를 고려하지 않는 조건에서 제도영역을 확보하

고, 부품의 최대 크기를 산출하여 형상을 확대한다. 확대된 상태에서 모든 직선과 원, 호의 길이, 반경, 등을 체크하여 상세도의 필요성을 검색한다. 상세도가 필요할 때 평면도는 상세도를 그리기 위해 축소된다. 축소된 상태에서 다시 상세도를 제도할 부분을 검색한다. 상세도를 그릴 부분의 수는 작은 요소들이 서로 근접해 있을 경우, 이들을 하나의 상세도에서 제도할 것인지, 개별 상세도를 만들어 제도할 것인지를 판단하여 결정된다. 이것을 결정하는 방법은, 우선 임의의 작은 요소하나에서 시작하여 그 요소를 포함하는 미세부분에 다른 작은 요소가 있는지 확인한다. 그 요소들에는 상세도 제도 범위가 결정되었음을 나타내는 기호가 부여된다. 이러한 방법으로, 모든 요소들에 대해서 소속될 상세도의 영역을 결정해주면, 결과적으로 개별 영역을 갖는 상세도의 수가 결정되고, 도면에서 상세도 영역과 위치를 결정할 수 있게 된다. Fig. 13은 펀치를 설계할 때 상세도가 자동으로 제도되어 출력된 도면이다. 이 부분은 설계자의 미적 추구에 대한 요구가 매우 강한 부분으로 설계자를 만족시키기 위하여 수차례의 기술회의를 하였고 프로그래밍하는 데 많은 시간을 요구한 부분이다.



Fig.13 Detail drawing of punch

#### 4. 결 론

자동설계프로그램은 I-Deas drafting을 기본 S/W로 사용하였으며, CAD 지원 프로그램과 C언어 와 motif를 이용하여 유닉스 환경에서 구현되었다. I-Deas Drafting 에 내장된 언어와 C 언어 사이의 데이터 공유를 위해 독자적인 형상정보에 대한 자료 구조를 개발하였고, 이 자료구조는 CAD 상의 설계 형상을 효율적으로 조정할 수 있는 수단으로 사용되었다. 금형의 부품도 설계부분이 금형설계에서 가장 많은 시간을 소요하기 때문에 이 부분은 일괄 처리방식으로 자동으로 수행할 수 있도록 하였다. 상용 CAD를 사용하여 금형 한 벌에 대한 설계 시

간은 45일이 소요되었으나, 금형자동설계프로그램을 적용할 경우 10일로 단축되었다. 금형제품도 설계부터 치수기입까지 설계자동화 함으로서 수동설계에서 발생하는 설계 오류가 방지되고 신제품에 대한 금형개발 기간을 단축하는 효과를 가져왔다. 특히, 자동설계프로그램을 적용했을 때 치수기입 오류는 발생하지 않았다. 이와 더불어 체계화된 설계표준에 의하여 설계한 금형은 품질이 안정될 뿐 아니라 그에 의해 생산된 리드프레임의 품질도 향상되었다.

#### 참고 문헌

1. Choi, J. C., Kim, B. M., Kim, C., Kim, J. H., Kim, C. B., "A Study on the Development of Computer Aided Die Design System for Lead Frame," Semiconductor, JKSP, Vol. 16, No. 6, pp. 123-132, 1999.
2. Choi, J. C., Kim, B. M., Kim, C., Kim, J. H., Kim, C. B., "Development of Progressive Die CAD/CAM System for Manufacturing Lead Frame, Semiconductor," JKSP, Vol. 16, No. 12, pp. 230-238, 1999.
3. Kim, Y. Y., "A Coarse Mesh Model for Numerical Analysis of Leadframe Deformation Due to Blanking Residual Stress," Submitted to JKSP, 2004.
4. Kim, Y. Y., Lee, D. H., "Local Buckling Analysis of the Punch in Stamping Die and Its Design Modification," JKSP, Vol.16, No. 3, pp. 25-29, 1999.
5. Hong, S. H., Han, S. Y., Kim, Y. Y., "A Study on Extraction and its Storage Method of Topological Information from Common 2-D CAD Using the Boundary-Representation Method," JKSP, Vol. 16, No. 9, pp. 25-33, 1999.
6. I-Deas Drafting Programmer's Guide, I-Deas MS2.1

**부록: 금형 설계 공정**

설계공정	세부설계	내 용
개념 설계	제품도수정	제품 형상치수는 소숫점 4째자리에서 반올림하여 형상 변경함. 치수누적오류 발생을 방지하기 위함.
	펀치인선부	수정제품도로부터 펀치인선 형상 설계.
	블랭킹순서	펀치인선 형상을 전개하여 블랭킹 순서, 금형의 전체길이, 금형 set 수, 변형 교정 펀치 등을 결정하여 각각 금형의 레이아웃 생성.
조립도 설계	펀치 조립도	punch holder, punch plate, punch backing plate를 설계. 펀치를 plate에 고정하기 위한 shank형상을 설계, plate에 펀치를 고정시켜주는 고정 plate 설계. pressure pin을 배치. 각 부품들의 체결형상 설계.
	스트리퍼 조립도	stripper holder, stripper plate 설계. 정밀한 펀치의 왕복운동을 가이드하기 위해 펀치와 적정클리어런스를 갖는 펀치 가이드형상 설계. 각 부품들의 체결형상 설계.
	다이 조립도	Die holder, die plate, die backing plate, die insert 설계. 리드프레임 소재의 이송을 유도하는 guide plate 설계. Die insert에는 펀치의 인선부 형상에 대해서 적정 클리어런스를 갖도록 설계. Die backing plate와 die holder에는 블랭킹 후 발생하는 스크랩(scrap)의 배출구와 각 부품의 체결부 설계.
부품도 설계	부품도 설계 (치수기입)	각 부품에 대한 부품도 설계. 치수기입, 측면도, 특정형상에 대한 가공방법, 등을 표기.
	CAM가공설 계	각 부품에 대한 NC 코드 생성용 도면을 설계하고 NC code를 생성.
	부품목록 작성	부품의 번호, 이름, 소재, 완성치수, 열처리 방법, 수량, 등의 내용을 포함한 부품 목록 생성.