

# 사출금형 몰드베이스와 몰드 금형 부품의 3차원 CAD라이브러리 구축에 관한 연구

이철수<sup>1</sup> · 박광렬<sup>2</sup> · 김용훈<sup>2</sup>

<sup>1</sup>전남대학교 산업공학과 / <sup>2</sup>전남대학교 산업공학과 대학원

## A Study on Implementation of 3D CAD Library for Injection Mold Base and Mold Components

Cheol-Soo Lee<sup>1</sup> · Gwang-Ryeol Park<sup>2</sup> · Yong-Hoon Kim<sup>2</sup>

Mold base design is one of the most critical steps in production of plastic parts. This paper presents an efficient method for generating a 3D CAD library for injection mold base and mold components. In this paper, a new description language (DL) formats are proposed. These DL formats can represent the geometric feature, the variable dimensions and the assembly structures of mold base and its components. And the processes of adding the mold components and creating the mold assemblies can be performed very easily and efficiently by using DL scripts. Proposed method has been tested by implementation of 3D library with DL scripts and DL-interpreter with C language. The DL and interpreter are independent of CAD system. They were applied to the automatic mold base designing system which works on Unigraphics V15. The system is developed with UG API function under Windows-NT environment.

### 1. 서론

#### 1.1 개요

최근 들어 제품의 외관은 그 제품의 경쟁력을 결정짓는 매우 중요한 요소가 되고 있다. 예를 들어 각 핸드폰 제조업체에서 새로운 디자인을 더 빨리 출시하는 능력은 시장을 선점하고 결국 경쟁에서 우위에 서는 능력의 기준이 되고 있다. 각종 통신 기기 및 가전 제품에 플라스틱 성형 제품이 많은 것을 고려할 때, 생산 공정에서 개발 기간을 단축할 수 있는 사출금형 자동 설계 시스템은 매우 중요한 의미를 갖는다. 실제로 몰드베이스 설계 방법은 매우 다양하며 형상에 따라 몰드베이스가 복잡한 구조를 가지는 경우도 많다(Rawin and Venkar, 1997). 본 연구에서는 사출금형 자동설계 시스템에서 이용될 수 있는 3D형상의 몰드베이스 및 몰드부품 라이브러리를 구축하기 위한 효율적인 방법을 제안한다. 제안된 방법은 형상 및 조립 관계 등을 CAD시스템과는 독립적인 스크립트 언어로 표현하고 있으며, 기존 3D 모델링을 지원하는 CAD플랫폼에 쉽게 적용될 수 있도록 설계되었다. 제안된 방법을 사용함으로써 자동 설계시스템의 개발이 용이해지며 따라서 사출금형을 설계할 때 표준부품에 대한 치수결정 등 설계작업에 요하는 시간을 대폭 줄일

수 있다. 또한 라이브러리에 표준화된 3D형상의 CAD데이터가 정확하게 저장되어 있으므로 설계에 보다 정확성을 기할 수 있으며(한국생산기술연구원, 1999), 솔리드 기반 CAD에 적용되도록 설계되어 기존 솔리드 모델러의 인터페이스를 통해 CAD 시스템의 데이터베이스와 연동으로 전체 형상 데이터 조작 및 관리도 용이하다.

#### 1.2 관련된 기존 연구

몰드베이스 자동설계를 2D기반에서 지원하는 상용 몰드베이스 자동설계 시스템에는 AutoCAD를 기반으로 개발된 CAD MAX(폴리테크, 1997)와 MOLDACE V3 등이 있다. 이처럼 2D기반으로 몰드베이스나 부품을 설계하는 경우 3D기반 시스템에 비해 형상에 관한 정확한 정보를 파악하기 힘든 경우가 있을 수 있다. 3D 솔리드 모델러를 기반으로 하는 시스템으로는 Unigraphics에서 작동되는 IMOLD(1999)와 MOLDWARE(1997), RAMDEX(1999) 등이 있고 Pro/ENGINEER용으로 Pro/MOLDESIGN(1999) 등이 있으며 Eureka97기반으로 한 K-MOLD(1999) 등이 있다. 3D기반으로 몰드베이스를 구축하는 경우 몰드 및 부품의 형상에 대한 라이브러리가 3차원 형상으로 구축되어 있어야 하는데 제조 회사마다 매우 다양한 규격을 가지고 있으며 비표준 형상도 있다. 3D 기반의 설계 시스템의 경우 조립 상태, 부품 명세

그리고 가공 정보 등의 산출이 용이해야 한다.

1.3 제안된 시스템의 특징

기존 2D기반 몰드베이스 생성 시스템에 비하여, 본 연구에서 제안한 시스템에서는 3차원으로 정의된 데이터베이스를 사용하기 때문에, CAM작업에 필요한 데이터를 직접 얻어내어 NC기계를 이용한 가공 자동화에 이용될 수 있고, CAE를 활용하여 치수결정, 설계 및 계산에 의해 표준 부품의 선택의 최적화 여부를 확인하는 기본 데이터를 제공할 수 있다.

본 연구에서는 몰드베이스 및 부품 라이브러리 정보를 3차원 CAD시스템의 라이브러리로 구축하기 위해 형상 정보를 기술하는 간어 언어(Description Language;DL)를 제안한다. 제안된 방법으로 F사(Fujitsu, 1997)의 대표적인 사출금형용 몰드부품 및 몰드베이스 정보를 라이브러리로 구축하였다. 특히 DL을 이용하여 형상 라이브러리의 구축이 용이해지고 라이브러리를 적용하여 실제로 구현한 몰드베이스 설계 자동화 시스템의 개발 과정이 매우 단축되었다. 기존의 3D기반의 시스템인 I-MOLD나 RAM-DES 또한 3차원 라이브러리가 구축되어 있으나, DL을 이용한 경우의 커스터마이징(Customizing) 수준을 제공하지는 못한다.

DL을 사용했을 때 얻게 되는 장점으로는 먼저 라이브러리의 확장 및 분야, 형태별 커스터마이징(Customizing)이 용이하다. 사출금형 생산업체의 특성상 커스터마이징이 반드시 필요하게 되는데, DL을 사용하면 프로그램을 수정하여 재컴파일할 필요 없이, 단지 DL 만을 수정함으로써 확장 및 수정이 가능하게 된다. 또한 CAD시스템과는 독립적인 DL해석(interpret)모듈이 구현되어 사용되므로, 현재의 플랫폼은 Unigraphics이나 CATIA,Pro/ENGINEER,AutoCAD/MDT 등의 타 플랫폼으로의 이식이 용이하다. 즉, 솔리드를 생성하는 부분만 이식될 CAD플랫폼이 제공하는 API(Application Programming Interface)를 사용하여 제작성하면 된다.

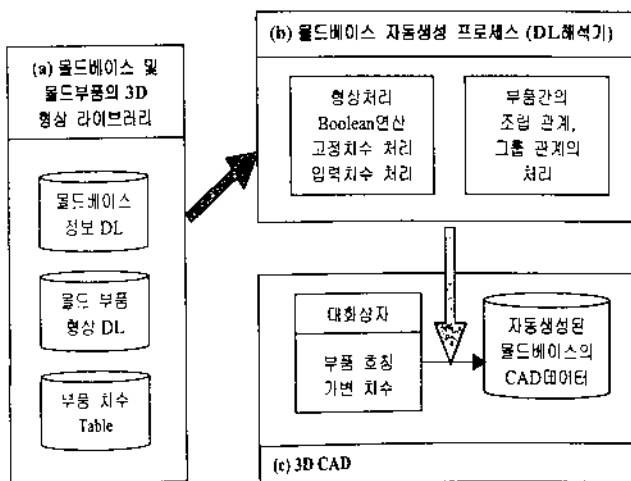


그림 1. 제안된 DL방식 부품 라이브러리를 이용한 몰드베이스 자동생성 시스템 개요.

2. 몰드베이스 부품의 형상 라이브러리

2.1 몰드베이스부품의 분류

몰드베이스 부품은 용도 등에 따라 여러 방법으로 분류 (Rawin and Venkat, 1997; 유병렬, 1990)한다. 본 연구에서 구축된 시스템에서는 F사의 몰드베이스 데이터(유병렬, 1990)를 기준으로 3차원 CAD 데이터베이스를 구축하였다. 라이브러리에는 크게 몰드부품과 몰드베이스가 생성하는 부분으로 구성되어 있다.

몰드부품은 가동반축과 고정반축과의 안내로 되는 가이드 핀 및 가이드부시, 돌출판(이체터 플레이트) 및 복귀용 리턴핀 등이 포함된다. 그러나 최근에는 사출 성형기용 금형의 경우, 스프루 부시나 로케이트링 등도 포함하는데 대표적인 부품을 <표 1>에서 보이고 있다.

표 1. 몰드베이스 부품 분류의 예

분 류	종 류
가이드 부품(Guide Elements)	가이드 핀, 가이드부시, 서포트핀 등
위치결정 부품 (Locate Elements)	로케이드 링, 스톱핀 등
디몰딩 부품 (Demolding Elements)	밀핀, 리턴핀, 앵글러핀 등
게이트 부품(Gate System Elements)	스프루 부쉬 등
기계요소 부품 (Machine Elements)	스프링, 아이볼트 등
냉각수 장치 시스템 부품(Cooling System Elements)	커플링 및 니플, O-링, 스크류 플러그 등

몰드베이스는 금형의 본체가 되는 것이며, 캐비티부를 구성하는 형판을 비롯하여, 각종 플레이트류가 주체로 되어 있다. 본 연구에서 라이브러리로 구축한 F사의 몰드베이스의 경우 2 플레이트 또는 3플레이트 타입 등의 몰드베이스를 제공한다. F사의 분류 상에서 각 플레이트 타입은 가이드나 밀핀, 서포트핀의 사양 등에 따라 각 시리즈별로 나누어지는데(기산, 1997), 분류된 표준 몰드베이스 종류의 예를 <표 2>에서 보이고 있다 (기산, 1997).

표 2. 표준 몰드베이스 규격의 예(F사 분류 방법)

Type	Series	
2 Plate type (side gate mold base)	S	SA,SB,SC,SD
	D/E	DA,DB,DC,DD,EA,EB,EC,ED
3Plate Type(Pin Point Gate Mold base)	F/G	FA,FC,DA,DC
	H	HA,HB,HC,HD

실제로 연가소성 플라스틱 사출 성형용 금형에는 구성 방법 등에 따라 여러 가지 타입이 있는데 어느 것이든 스프루나 러너, 세이트 등 용융재료가 흐르는 통로부분의 구조에 따라 각각의 특징이 있다. F사가 분류한 2 플레이트 타입과 3 플레이트 타입을 구별하는 기준도 이러한 유동부분의 구조에 의한 것이다(기신,1997).

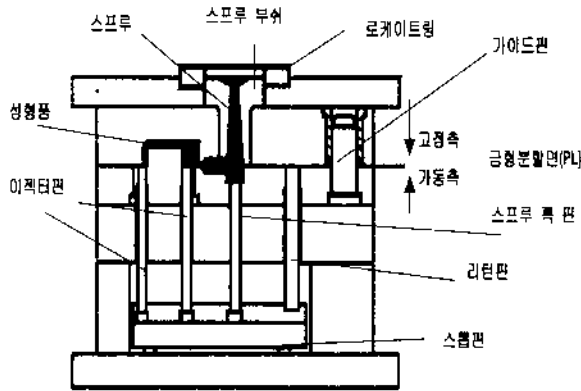


그림 2. 2 Plate Type Mold Base의 구성 예.

현재 실용되고 있는 사출 성형용 금형의 대부분이 2 플레이트 타입에 속하는 경우가 많은데 <그림 2>와 같이 러너나 게이트가 금형의 고정축(사출 성형기의 고정축 다이플레이트에 부착하는 부분)과 가동축(형조임 실린더에 직결한 가동축 다이플레이트에 부착하는 부분)과의 분할면 위에 있는 것이 특징이다. 이와 같이 금형의 주요 부분이 고정축(보통의 경우 캐비티 축이 된다)과 가동축(코어축)과의 2개 부분으로 이루어져 있기 때문에 2 플레이트형이라고 불린다(기신, 1997).

3플레이트 타입의 몰드베이스는 제품의 분할면(PL)과는 별도로 또 하나의 평면에 러너가 배치되어 있어서 형을 열 때마다 이 면을 열고 러너를 빼내는 형식이다. 이 타입은 코어부분 및 캐비티 부분으로 되는 2개의 형판과 그밖에 러너를 떼어내기 위한 러너 스트리퍼 플레이트(Runner Stripper Plate)라는 1개의 플레이트가 삽입되어 있는데, 몰드베이스의 주요부분이 이들 3개의 플레이트로 구성되어 있기 때문에 3 플레이트 타입이라 한다. 고정축 부착판은 사출 성형기의 고정축 다이플레이트에 부착하는데, 고정축의 형판과 러너 스트리퍼 플레이트와는 고정축 부착판에 설정된 긴 가이드핀(서포트 핀)이 위를 섭동하게 되어 있다(기신,1997).

2.2 몰드베이스 부품 라이브러리

실제로 많은 다양한 형태의 표준화된 몰드베이스 부품의 규격과 치수가 있지만 비표준 규격의 치수가 필요할 수 있고 새로운 형상의 부품이 요구되는 경우도 있을 수 있다. 전자의 경우 새로운 치수의 형상을 지원하는 라이브러리 시스템은 그 형상의 각 치수호칭에 대해 단순한 치수 테이블을 보관하는 DB에 치수만 추가하면 되지만 후자의 경우 금형 자동 설계 시스템이 그 형상을 지원하지 않는다면 그 부품은 사용할 수 없게

된다. 물론 충분히 여러 가지 모양의 형상 라이브러리를 구축하면 되겠지만 매우 비효율적인 라이브러리가 구축되게 된다. 따라서 사출 금형 자동 설계 시스템과는 독립적으로 갱신될 수 있는 몰드베이스의 부품 라이브러리 구축이 필요하다.

몰드베이스는 독립적으로 관리될 수 있는 부품형상과 달리 조립과 관련된 그룹화가 필요하고 플레이트 등의 몰드베이스와 부품간의 위치들은 고정된 치수보다는 비례 관계 등으로 지정되는 경우가 많다. 필요한 치수를 모두 계산하여 몰드베이스 관련 정보를 라이브러리로 구축할 수도 있겠지만 이 경우 역시 몰드베이스 정보의 추가할 때 매우 번거로운 작업이 발생한다. 본 연구에서는 이러한 몰드베이스 및 부품 형상을 정의하고 몰드베이스 자동 생성 정보를 나타낼 수 있는 DL에 의한 라이브러리 구축 방법을 제안한다.

2.3 부품 형상 정보의 기술언어(DL)

본 연구에서 제안하는 DL은 형상을 정의하기 위한 여러 가지 기능을 가지고 있다. 먼저 가장 기본적인 기능은 실린더, 블록, 구 등의 단순한 솔리드 형상을 정의할 수 있는 기능이다. <그림 3>의 (a)는 이러한 기본형상(primitive feature)을 정의하는 스크립트의 예이고 (b)는 생성된 실제 형상의 모양이다. 스크립트의 구문 중에서 BLK는 직육면체, CYL은 실린더 형상, SPHER은 구, CONE는 원뿔 또는 원뿔대 형상이며 W (width), L (length),

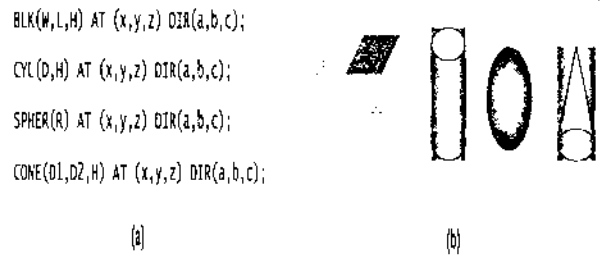


그림 3. 제안된 DL에 의한 기본 솔리드 형상 정의 예.

D(diameter), R(radius), H(height) 등은 치수이다. AT구문 다음에 있는 (x,y,z)는 형상이 위치할 좌표이며 DIR구문 뒤의 (a,b,c)는 형상의 방향 벡터이다. 형상 정의를 위한 두 번째 기능은 3D 형상들간의 부울 연산 기능이다. 이러한 기본형상을 union 및 subtract 기능에 의해 형상을 정의하면 복잡한 형상도 CSG

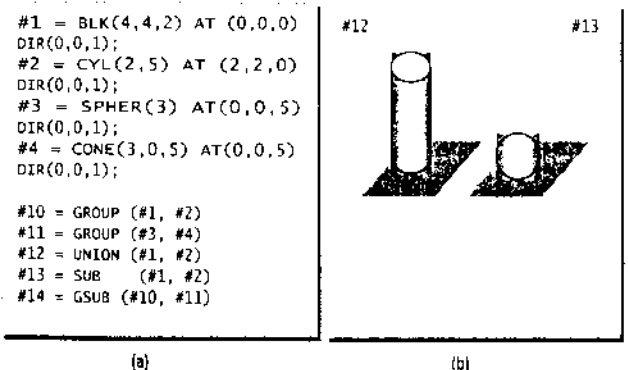


그림 4. 제안된 DL에 의한 기본 솔리드 형상의 정의의 예.

(Constructive Solid Geometry)방식으로 트리(tree)를 구성해 생성해낼 수 있게 된다. <그림 4>의 (a)는 실제 치수를 대입한 경우의 기본 형상과 부울연산을 하는 스크립트의 예이며 (b)는 이 스크립트에 의해 생성되는 실제 형상 몇 가지를 보인 것이다. 예를 들어 SUB와 UNION은 각각 subtraction, union 연산을 의미하고 GSUB는 GROUP분으로 묶어진 형상들을 하나의 형상으로 인식하여 subtraction 연산을 수행해 준다. 이처럼 DL은 스크립트 문법 자체가 형상 정보에 대한 직관을 주기 때문에 부품의 모양뿐 아니라 새로운 몰드베이스 부품 형상이나 관련 정보의 추가가 매우 용이하다

몰드 부품 형상의 정의는 위의 기능만으로 가능하지만 몰드베이스에서는 각 부품의 조립 및 그룹 관계를 가지고 있어야 한다. 따라서 플레이트 등의 솔리드 형상과 부품 형상들의 그룹화와 어셈블리 구조의 표현 기능이 필요하다. 그 외에 형상 관계 정의가 용이하도록 판단과 부합수 호출기능 등을 추가로 구현하였다. 마지막으로 각 형상 정의 블록에는 가변 치수를 지원하기 위한 인자(parameter)전달이 있고 블록의 시작과 끝의 구분자 그리고 판단문과 주석문 등이 표현 가능하도록 하였다. 이처럼 몰드베이스를 기술한 DL 스크립트는 형상 정의 부분과 복잡된 몇 개의 추가된 구분으로 기술되는데 다음 장의 몰드베이스 자동 생성 프로세스 설명 과정에서 예시된다.

2.4 치수 테이블

DL에 의해서는 몰드베이스 부품의 모양과 조립 등의 관련 정보가 정의된다. 실제 치수에 관한 정보는 별도의 치수 테이블로 관리한다. 따라서 같은 모양과 같은 용도로 이용되는 새로운 치수의 부품은 치수만을 추가하면 된다. 부품에는 밀핀과 같이 임의의 길이 입력이 필요하고 나머지 치수만 고정된 경우가 있다. 이 경우 길이는 자동 설계 과정의 대화상자에서 치수를 입력받을 수 있어야 한다. <그림 5>의 (a)는 밀핀의 형상 정의 DL의 예인데 이 타입의 밀핀은 D1,D2,H1,H2 등 4개의 치수가 필요하다. <그림 5>의 (b)는 치수 테이블에서 밀핀 부분을 보인 것인데 여기서 H4에 해당하는 @는 고정된 치수가 아닌 작업자가 입력하는 치수인 경우의 표식이다. 실제로 <그림 5>의 (c)와 같은 대화 상자에서 @값을 입력받는다.

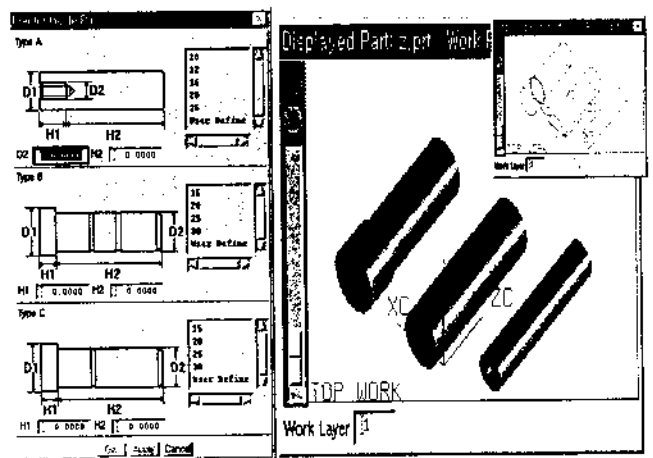
<그림 5>의 (c)에서 보듯이 몰드베이스 자동생성 과정에서는 몰드베이스 설계자가 평소에 익숙한 부품의 호칭으로 여러 가지 치수의 부품을 대화상자를 통해 선택한다. <그림 5>의 (b)의 치수 테이블에서 괄호 안의 숫자는 밀핀가이드 부품의 형상에 종속된 치수의 호칭이다. 실제로 치수를 입력받을 때는 치수에 관한 직관을 갖도록 형상의 그림(image)을 표시하는 것이 좋는데 <그림 5>의 (c)에서 그 예를 보이고 있다. 이처럼 하나의 부품 형상을 생성하기 위해서는 <그림 5>의 (a),(b),(c)의 예처럼 형상 정의의 DL, 치수 테이블 그리고 몰드베이스 생성 과정에서 입력받은 치수 정보가 모두 필요하다.

3. 몰드베이스 자동 생성 시스템

```

@ EJECTOR GUIDE PIN
@a
PARAM (D1,D2,H1,H2);
#1 = CYL(D1,H1+H2) AT(0,0,0) DIR(0,0,1);
#2 = CYL(D2,H2) AT(0,0,0) DIR(0,0,1);
#3 = SUB(#1,#2);
#3 = COLOR WHITE;
END; /*-----*/
@b
PARAM (D1,D2,H1,H2);
#1 = CYL(D1,H1) AT(0,0,0) DIR(0,0,1);
#2 = CYL(D2,H2) AT(0,0,H1) DIR(0,0,1);
#3 = UNION(#1,#2);
#3 = COLOR WHITE;
END; /*-----*/
@C
PARAM (D1,D2,H1,H2);
#1 = CYL(D1,H1) AT(0,0,0) DIR(0,0,1);
#2 = CYL(D2,H2) AT(0,0,H1) DIR(0,0,1);
#3 = UNION(#1,#2);
#3 = COLOR WHITE;
END;
    
```

(a) (b)



(c) (d)

그림 5. 형상 라이브러리(형상, 치수)와 호칭 입력에 의한 몰드부품(밀핀가이드) 생성의 예.

3.1 시스템의 구성

몰드베이스 부품 라이브러리가 구축되면 몰드베이스 생성 프로세스와 사용자 인터페이스에 의해 몰드베이스 자동 생성 시스템을 구축할 수 있다. <그림 6>의 (a)는 몰드베이스 부품 라이브러리어며 <그림 6>의 (b)는 CAD작업자가 선택한 유형의 몰드베이스 전체를 생성해주는 프로세스 모듈인데 형상 라이브러리와 CAD시스템과는 독립적이다. <그림 6>의 (c)는 적용되는 CAD시스템의 API에 의존하여 구현된다.

시스템을 이용하는 작업자 측면에서 데이터의 큰 흐름은 <그림 6>에서 보면 ①②③으로 표시된 모듈을 지나가게 된다. 작업자는 먼저 ① 사용하는 CAD시스템을 이용해 구현된 사용자 인터페이스에서 원하는 몰드베이스 시리즈, 규격 등과 추가될 몰드 부품의 호칭, 가변 치수 등을 입력해준다. 이 정보를 이용하여 몰드베이스 부품 라이브러리(DB)에서 몰드베이스 생성에 필요한 실제적인 데이터를 꺼내어 별도의 메모리에 보관한다. 이 데이터는 ② DL로 표현된 라이브러리의 내용을 해석하여 치수와 형상, 구성에 관한 것들을 모두 반영한 몰드베이스 전체의 정보를 모두 가지고 있다. 몰드베이스 정보란 형상 정

보 외에도 조립이나 그룹과 관련된 BOM정보도 포함하게 되는데, ③이 내용을 작업자가 사용하는 CAD시스템의 데이터베이스에 추가하면 몰드베이스가 완성된다. 본 연구에서는 3D 솔리드 모델링을 지원하는 CAD시스템을 기반으로 DL을 작성하게 되어 있다. 따라서 CAD시스템은 솔리드 생성 API가 지원되어야 하고 사용자 정의 인터페이스도 API수준에서 지원되어야 한다.

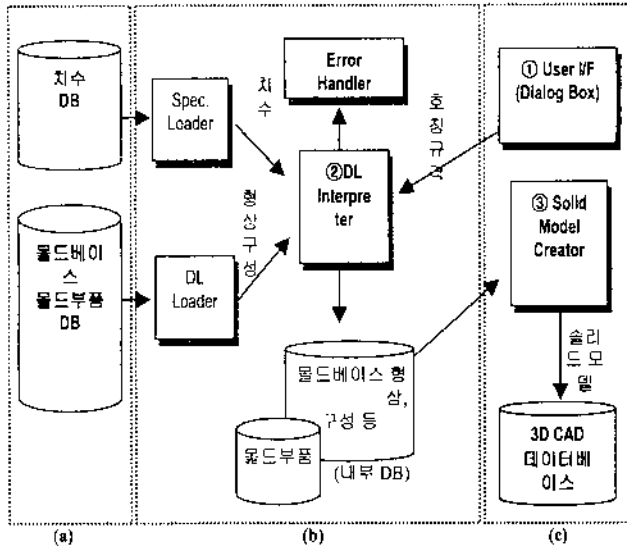


그림 6. 제안된 몰드베이스 자동 생성 시스템의 기능별 module 및 data flow.

3.2 몰드부품 생성 프로세스

<그림 6>에서 형상 라이브러리에 해당하는 (a)는 치수 테이블과 몰드부품 DL스크립트 형태로 표준 및 비표준부품을 생성할 때 이용되는 부품 DB이다. 치수 데이터베이스에는 각종 몰드 부품과 몰드베이스의 3차원 치수데이터 파일이 들어 있고, DL 파일에는 몰드부품과 몰드베이스를 묘사한 DL스크립트 파일 등이 들어 있다.

이 파일을 해석하고 실제의 부품 생성작업을 수행하는 기능이 필요한데 <그림 6>의 (b)에 있는 DL Interpreter 모듈이 담당한다. <그림 6>의 (b)에서 규격과 치수 파일을 읽는 Spec. Loader와 DL스크립트 파일을 읽는 DL Loader는 이러한 형상 라이브러리 파일에서 DL Interpreter모듈이 필요한 정보만을 꺼내는 기능을 담당한다. 여기서 필요한 정보란 작업자가 선택한 몰드베이스에 따라 자동으로 결정되는 기본 몰드부품과 추가로 선택한 몰드 부품 등이다.

작업자의 선택은 <그림 6>의 (c)에 있는 User I/F 모듈에서 대화상자(Dialog Box)를 통해 입력받는다. 이 모듈은 <그림 6> 전체에서 보듯이 몰드베이스 부품 라이브러리 또는 몰드베이스 생성 프로세스와는 독립적인 모듈이다.

3.3 몰드베이스 생성 프로세스

몰드베이스 생성 프로세스는 위 절의 부품 생성 방식과 큰 흐름은 같이 한다. 하지만 DL Interpreter모듈에서 Solid Model Creator모듈로 전달되는 데이터에 몰드베이스의 BOM정보와 몰드베이스의 뼈대를 이루는 각 플레이트들과 각 플레이트에 심어지는 부품 사이의 위치 및 부울연산 정보 등이 추가된다. <그림 7>은 몰드베이스를 기술한 DL스크립트의 예를 보이고 있다.

```

@ SIDE GATE MOLDBASE
/*-----*/
/* 2_PLATE_TYPE */
/*-----*/
@ [SA]
PARAM(X,Y,A,B,C,U,GUIDE_SPEC,EJ_P_SPEC,T,S,E,F,L,LX,RPX,HBX,GPX,DHB
,CX,EPX,IBX,IBY,
GPY,RPY,DHBY,HBX,D,DHB,GP,HB,RP,GPH1,GBH1,DHBH1,DHBL,HBH1,HBL,RPH1,
B_NUM);
/* BOTTOM_PLATE */
#1 = BLK(LX,Y,L) AT((X/2)-(LX/2),0,0) DIR(0,0,1);
#2 = CYL(D,L) AT(X/2,Y/2,0) DIR(0,0,1);
#P1 = SUB(#1, #2);
#P1 = COLOR GRAY;
/* SPACER_BLOCK1, SPACER_BLOCK2, EJECTOR_PLATE */
(b)IF(EJ_P_SPEC = 0){
/* EJECTOR_RETAINER_PLATE */
#P5 = BLK(EFX,Y,E) AT(CX+2.0,L+P) DIR(0,0,1);
#P5 = COLOR RED;
}
/* SUPPORT_PLATE */
/* CORE_PLATE */
/* CAVITY_PLATE */
/* TOP_PLATE */
/*-----*/
/* ASSEMBLE PARTS */
/*-----*/
IF (GUIDE_SPEC = 0) {
#S1 = CALL (GUIDE_PIN,A,GP) UID(A+S+B-GPH1) AT ((X/2)-(GPX/2),
(Y/2)-(GPY/2),
L+C+U) DIR(0,0,1);
}
(c)#S4 = CALL (HEXA BOLT,A,DHB) UID(L+C+B-4-(DHBH1-DHBL)) AT (0,0,0)
DIR(0,0,1);
(d)#NN = SAVE #S4 HB_L;
IF(HB_NUM=2){
(a)#B25= LOAD HB_L_M10 #S3 AT((X/2)-(DHBX/2), Y/2, 0) DIR(0,0,1);
#B26= LOAD HB_L_M10 #S3 AT((X/2)+(DHBX/2), Y/2, 0) DIR(0,0,1);
}
END;
  
```

그림 7. 몰드베이스 부품 라이브러리 중에서 몰드베이스 DL스크립트의 예.

이렇게 몰드베이스를 생성하는 과정에는 표준부품을 생성하는 과정이 포함되므로 몰드베이스 DL 블록을 해석하는 중에 표준부품 DL 블록을 호출하는 과정이 들어간다. 실제로 DL Interpreter모듈은 표준부품을 생성하는 DL스크립트와 몰드베이스를 생성하는 DL스크립트를 구분하여 해석하지는 않는다. 3.2절과 같이 부품을 생성하는 DL은 주로 솔리드 형상의 모양과 치수 등에 대한 단순한 스크립트인데 비해 몰드베이스 DL은 보다 복잡한 스크립트인데 그 내용에는 형상 정보 외에도 조립상태와 필요한 부품들의 그룹관계 등을 가지고 구조적인 생성 프로세스를 내포하고 있게 된다.

3.4 몰드베이스와 부품간 어셈블리와 위치 정보의 표현

몰드베이스는 종류에 따라 관련 부품의 형상과 위치가 결정된다. 몰드부품의 종류와 위치 및 형상 치수는 몰드베이스에 대한 치수와 관련되어 있는데 DL에서 보면 이러한 기준 치수들은 <그림 7>의 (a)부분과 같이 PARAM이하의 괄호 안의 매

개변수로 전달받는다.

실제로 DL에서 몰드 부품들의 형상과 위치를 정의하는 예를 <그림 7>의 (b)부분에서 볼 수 있는데 먼저 몰드베이스의 타입을 구분하는 판단문(IF)과 기준 치수를 이용한 수식(expression)으로 몰드부품의 형상과 치수 등을 제약하고 있다. 이와 같은 방식으로 몰드베이스에 따라 파라메트릭하게 결정되는 몰드부품의 형상 및 위치 등을 정의할 수 있다.

몰드베이스에는 같은 몰드부품이 위치는 다르면서 반복적으로 사용되는 경우가 많다. 이러한 경우의 처리를 위해 DL에서는 이미 한번 정의된 형상은 부함수 호출(CALL)에 의해 재사용할 수 있도록 해준다. DL에서는 <그림 7>의 (c)와 같은 방법으로 위치(position)와 자세(orientation)를 정의하는데 이 형상을 <그림 7>의 (d)와 같이 SAVE구문으로 형상 이름과 함께 저장할 수 있다. 저장된 형상은 DL을 기술할 때 LOAD구문으로 부품을 정의하는 단계와 위치 등으로 어셈블리 관계가 정의될 수 있다. <그림 7>의 (e)에서 이미 저장된 부품을 불러들이면서 위치와 자세를 결정하는 예를 보이고 있다. 이들 정보는 BOM 정보로 활용된다.

3.5 생성된 부품의 CAD데이터베이스 추가

부품 데이터가 저장되는 자료 구조의 예를 <그림 8>에서 보이고 있다. <그림 8>에서 보듯이 각각의 부품들은 하나의 엔티티로 간주되고, 이 엔티티에 대한 부품의 이름(entity\_name)과 CAD시스템의 데이터베이스에서 해당 부품을 인식하는 ID인 entity\_tag를 보관하고 나머지는 형상 라이브러리에서 얻어지는 정보를 보관하는 변수이다.

```
typedef struct _entity{
char entity_name[20];
tag_t entity_tag;
double origin[3];
double dir[3];
int color;
} entity;
```

그림 8. 몰드부품을 내부 데이터베이스에 보관하는 자료 구조의 예.

4. 적용사례

4.1 개발환경

본 연구에서 개발된 몰드부품/몰드베이스 3D CAD 라이브러리는 Windows NT Workstation 4.0과 Unigraphics(UG) V15.0을 기반으로 만들어졌으며, C 언어를 사용하여 MSVC V6.0에서 컴파일되었다. 사용 API는 Unigraphics에서 제공하는 UG/Open API 등(Electronics..., 1998)을 사용하였다.

4.2 표준 및 비표준부품 생성 과정

부품의 생성 과정은 <그림 9>와 같은 과정을 따른다. 표준 부품을 생성시킬 때에 CAD작업자가 입력해야 할 치수가 추천

값과 함께 디스플레이되고, 유저는 이 추천값을 사용하거나 수정할 수 있다. 비표준부품을 생성시킬 때에는 부품에 대한 모든 치수값을 유저가 입력해야 한다.

이처럼 CAD(Unigraphics)시스템에서 대화형으로 몰드부품을 생성할 때 부품의 종류 및 규격의 선택이 매우 용이해야 한다. 제안된 시스템에서는 <그림 9>와 같은 대화상자가 왼쪽에서 오른쪽으로 가는 방향으로 차례로 나타나 선택을 입력받는다. <그림 9>의 생성 과정에 따라 생성된 각기 다른 타입과 규격을 가지는 여러 가지 부품들의 예를 <그림 10>에서 보이고 있다.

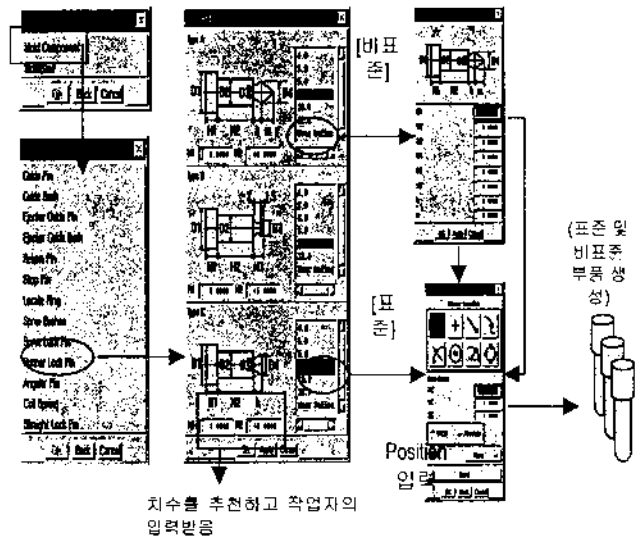


그림 9. 표준/비표준부품 생성 과정.

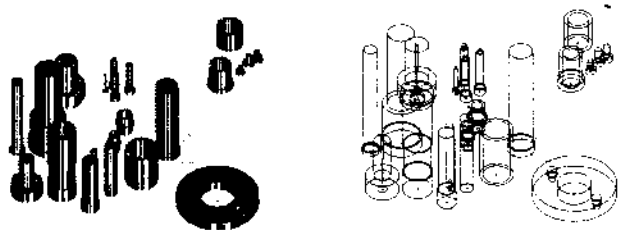


그림 10. Unigraphics 상에 생성된 표준/비표준 부품들.

4.3 볼드베이스의 생성

몰드베이스의 생성 과정은 <그림 11>과 같은 과정을 거친다. 몰드베이스 설계자는 CAD시스템에서 대화상자를 이용하여 원하는 몰드베이스 타입과 시리즈를 결정한 후 타입에 종속된 호칭과 치수를 선택한다. <그림 11>에서 대화상자를 이용하여 몰드베이스 설계 과정을 보이고 있는데 A, B, C 등과 같은 특정 부위의 치수를 선택할 수 있다. 필요한 값들이 대화상자를 통해 입력되면 DL interpreter 모듈은 데이터베이스에 구축된 치수 정보를 사용하여 몰드베이스를 구성하는 솔리드 형상을 생성하도록 한다. 솔리드 모델의 생성은 UG API로 구현된 Solid Model Creator모듈이 담당한다. Unigraphics 시스템에서 이와 같은 과정을 거쳐 자동 생성된 몰드베이스의 예를 <그림 12>에

서 보이고 있다.

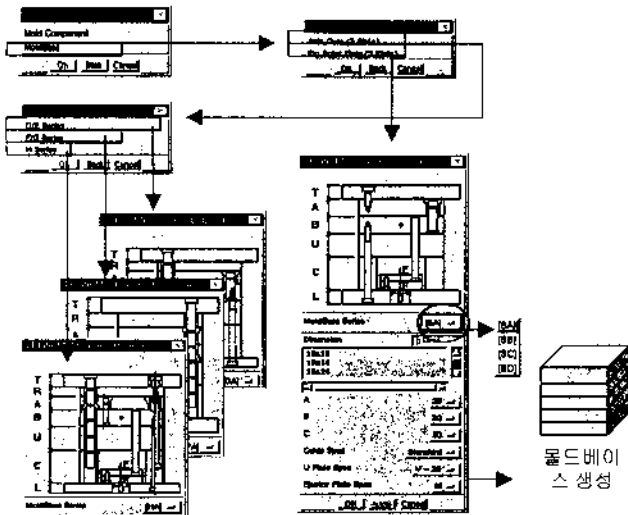


그림 11. 대화상자를 이용한 표준 몰드베이스 생성 프로세스의 예.

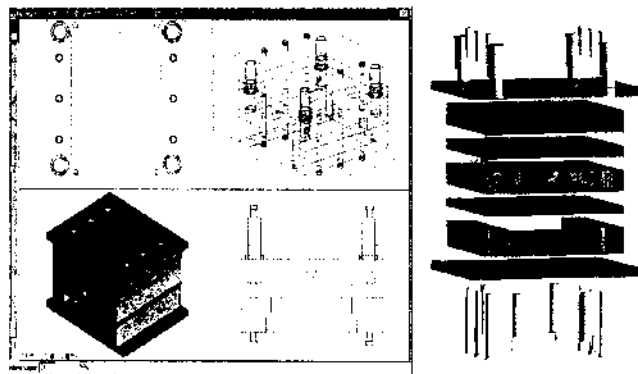


그림 12. Unigraphics 상에 생성된 표준 몰드베이스의 예.

5. 결론



**이철수**  
 1984년 한양대학교 산업공학과 학사  
 1986년 한국과학기술원 산업공학과 석사  
 1990년 한국과학기술원 산업공학과 박사  
 현재: 전남대학교 산업공학과 부교수  
 관심분야: CAD/CAM과 CNC 컨트롤러



**박광렬**  
 1993년 전남대학교 산업공학과 학사  
 1997년 전남대학교 산업공학과 석사  
 현재: 전남대학교 산업공학과 박사과정  
 관심분야: CAD/CAM, 공장자동화

본 연구에서는 3D형상의 몰드부품 형상과 몰드베이스의 BOM정보를 수용할 수 있는 CAD용 몰드베이스 라이브러리 구축 방법을 제안하였다. 특히, 형상을 기술하는 DL스크립트를 이용하는데 몰드베이스 및 부품 라이브러리의 확장과 분야 또는 형태별 커스터마이징이 매우 용이하다. DL로 구축된 형상 라이브러리와 실제 라이브러리의 모델을 생성해주는 DL interpreter.모듈은 CAD시스템과는 독립적이기 때문에 다른 CAD플랫폼으로 쉽게 적용될 수 있다.

제안된 형상 라이브러리 구축 방법을 이용하여 Unigraphics V15 CAD플랫폼에서 사출금형용 몰드베이스 자동 설계시스템을 실제로 구현하였다. 제안된 방법은 몰드베이스 자동 설계시스템의 개발을 용이하게 해주며 프레스 금형의 데이터베이스화에도 응용될 수 있을 것이다. 몰드베이스 부품의 3D Library를 이용한 자동설계 시스템이 구축되면 사출금형의 설계와 제작에 필요한 기간이 크게 단축되는 효과를 얻을 수 있으리라 기대된다.

참고문헌

이건우 외(1999), 상용 Solid Modeller 기반 사출금형 설계 CAD system의 개발, 99 한국CAD/CAM 학회 학술발표회 논문집, 199-203.  
 폴리테크 CAD/CAM Lab(1997), CADMAX Manual, 폴리테크.  
 한국생산기술연구원(1999), 기술개발성공사례, 한국생산기술연구원 48~49.  
 ㈜기신(1999), 기신-FUTAVA-몰드베이스, ㈜기신.  
 Korea CAD/CAM Service(1999), K-MOLD 제품소개서, Korea CAD/CAM Service.  
 柳柄烈(1990), 알기쉬운 플라스틱 금형, 성안당.  
 Electronic Data Systems Corporation Unigraphics Division(1998), UG/OPEN API Programming and reference Guide, EDS Corp.  
 Fujitsu Ltd(1997), MOLDWARE CAD user manual (Unigraphics), Fujitsu Ltd.  
 Manusoft Plastic PTE Ltd.(1999), The Intelligent and Assembly system(IMOLD), <http://www.eng.nus.sg/imold>.  
 Parametric Technology Corporation(1999), Pro/MOLDESIGN, <http://www-ptc.com>.  
 Rawin Raviwone and Venkat allada (1997), Artificial Neural Network Based Model for Computation of Injection Mould Complexity, *The Int'l Journal of manufacturing Technology*, 13, 577-586.



**김용훈**  
 1998년 전남대학교 산업공학과 학사  
 현재: 전남대학교 산업공학과 석사과정  
 관심분야: CAD/CAM, 공장자동화