

자바스크립트를 이용한 웹기반 공학계산 프로그램 개발

°진길주*, 꺾문규**, 허석*

Development of Web-Based Engineering Calculation Program Using JavaScript

Gil Ju Jin, Moon Kyu Kwak, Seok Heo

ABSTRACT

This paper is concerned with the development of web-based engineering calculation program using JavaScript. Recently, various techniques are developed based on the advance of the internet environment. The JavaScript can be used in the client PC without any interpreter which is necessary in Java. Simple formula can be easily constructed using the JavaScript and the results can be readily available without any playing. In this study, we demonstrated the method of constructing engineering web useful for engineers

1. 서론

인터넷이 등장한 이래 그 유용함은 이미 입증되었고 지금도 다양한 분야로 계속 그 영역을 확장해 나가고 있다. 초기 인터넷은 군사적인 연구 지원이 목적이었지만 지금은 교육, 의료, 오락, 방송 등의 모든 분야에서 일정 영역을 확보하고 더욱 새로운 응용 분야로 나아가고 있다.

교육분야의 경우 그 활용도는 비약적으로 발전하고 있는데, 기존의 지면을 통한 단순 학습에서 벗어나 직접 자료를 검색하고 다양한 시각효과 등을 통하여 그 학습 효과를 증대시키고 있다.

인터넷 프로그래밍 언어도 많은 변천을 거쳐 현재는 Java에 이르고 있다. 그러나 컴퓨터를 공부하

지 않은 사람들에게 인터넷 프로그래밍의 습득은 쉽지 않은 과제이다. 따라서 단순히 HTML 문장에 기능을 추가해서 Interactive 효과를 낼 수 있도록 JavaScript라는 언어가 만들어 졌다.

JavaScript는 얼마 전 까지만 해도 넷스케이프사의 LiveScript라는 이름으로 불려졌지만, 썬 마이크로시스템사와 넷스케이프사가 공동으로 언어를 만들기로 합의, 이름을 JavaScript로 부르게 되었다. 따라서 Java라는 이름과 유사하기 때문에 Java를 쉽게 만든 것이라고 생각하는 사람도 많지만 많은 면에서 차이가 있다. Fig. 1에서 보면 Java는 인터넷 웹 브라우저에서 들어가는 애플릿(웹 페이지 내의 작은 공간에서 실행되는 Java 미니 응용 프로그램)을 만드는 것이 주목적이고, JavaScript는 웹 브라우저를 제어하는 것이 주목적이다.[1]

*동국대학교 기계공학과 대학원

**동국대학교 기계공학과 교수

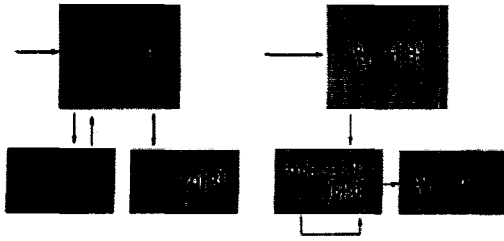


Fig. 1. Java와 JavaScript 흐름도

JavaScript의 특징을 살펴보면 먼저 HTML 문서 안에서 자바스크립트로 기술된 프로그램을 넣어둠으로써, 바로 실행할 수 있게 한다. 또한 CGI (Common Gateway Interface)에서 사용되는 Form 인터페이스와 연결할 수 있다. button form에서 button을 누르면 JavaScript의 어떤 함수가 실행 되도록 미리 설정해 놓을 수 있다. Table 1을 보면 Java에서는 클래스를 선언하고 method를 만든다. 그리고 메모리를 할당하고 method를 호출하는 복잡한 절차를 거치게 되는데 반하여, Java Script에서는 미리 선언할 필요 없이 필요할 때마다 method를 객체에 달아 두어 쓸 수 있다.

Table 1. Java와 JavaScript의 비교

JavaScript	Java
클라이언트에 의해 interpret	서버에서 컴파일, 클라이언트에 의해 interpret
객체 기반 언어 (클래스개념, 상속개념 없음)	객체 지향 언어 (클래스개념, 상속 개념)
HTML내에서 같이 기술됨	HTML페이지내의 작은 공간에서 로딩 되어서 별도로 수행됨
미리 변수형 정할 필요 없음	미리 변수형 정해야 됨

본 연구에서는 이러한 JavaScript와 급격하게 성장하고 있는 인터넷 환경을 이용하여 인터넷에서 공학용 계산을 할 수 있는 환경을 구축하였다. 일반적인 공학 계산은 공학용 계산기나 공학용 컴퓨터 프로그램을 이용하여 계산하는 것이 일반적이지만 본 연구에서는 인터넷의 홈페이지에 접속함으로써 계산을 할 수 있게 구현하였다.

본 연구에서 개발된 웹 기반 공학용 계산 프로그램은 언제 어디서나 쉽게 사용할 수 있어 많은 공학도의 사용이 기대된다.

2. 공학계산 프로그램 시스템의 설계

JavaScript를 이용한 공학계산은 서버에서 interpret를 해야되는 서버 언어와는 달리 인터넷을 통해 사용자가 접근하면 별도의 interpret과정 없이 실행할 수 있다. 공학계산 프로그램 시스템의 모형은 Fig. 2과 같다.

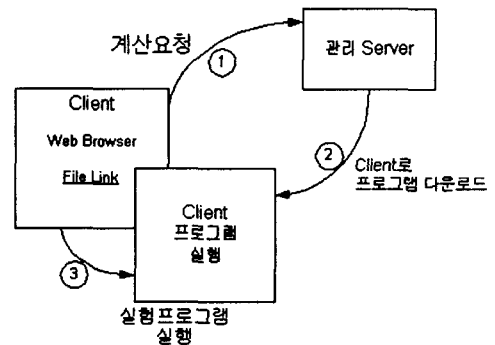


Fig. 2. 공학계산 시스템 모형

공학계산 수행 과정은 다음과 같다. 우선 클라이언트가 웹 상에서 계산 요청을 하면(①) 관리서버에서 클라이언트로 프로그램을 다운로드 한다. (②). 클라이언트는 입력값을 가지고 클라이언트 프로그램을 실행하여 계산한다(③). 따라서 웹 기반 공학계산 프로그램의 개발을 위해서는 클라이언트에 크게 부담을 주지 않으면서 쉽게 실행될 수 있는 환경이 필요하다. 이런 점에서 JavaScript가 적합하다고 말할 수 있다.

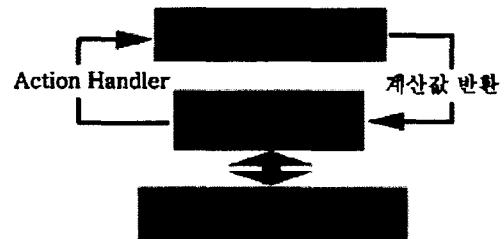


Fig. 3. HTML내의 JavaScript 흐름도

```

<html>
<head>
<title> 제목 </title>
<script language="JavaScript">
<!--
function 함수이름 (object)
{ with (Math)
{
var 매개변수1 = object.매개변수1.value;
var 매개변수2 = object.매개변수2.value;

var 매개변수3 =계산식1;
object.매개변수3.value = 매개변수3;
var 매개변수5 =계산식2;
object.매개변수4.value = 매개변수4;
} }
// --> </script>
</head>
:
<form name="form1">
<input type="입력값형식" name="입력값이름" value="초기값"
OnKeyUp="함수이름(this.form)">
:
<input type="출력값형식" name="출력값이름" size="크기">
</form></body>
</html>

```

Fig. 3은 HTML내에서 JavaScript와 HTML사이의 관계를 나타낸 흐름도 이다. 위의 소스는 공학 계산 프로그램 개발에 사용된 HTML과 JavaScript의 전반적인 개요이다. JavaScript는 HTML의 head 부분에 삽입되어 실행된다. 그리고 HTML의 입출력구문에서 입출력값의 형식, 이름, 그리고 실행방법을 정의하였다.

Fig. 4은 공학계산 프로그램이 링크되어 있는 홈페이지의 모습이다. 클라이언트는 사용하고자 하는 프로그램으로 들어가 기본적인 데이터 정보만을 입력하면 된다.

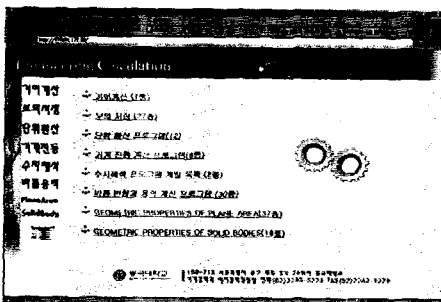


Fig. 4. 홈페이지

3. JavaScript 프로그램

전체 개발한 프로그램의 일부만 소개하고 나머지는 홈페이지를 방문하여 볼 수 있다. 현재 제공되는 계산 프로그램은 다음과 같다.

- * 기어 계산 (7종)
- * 보의 처짐 (27종)
- * 단위 환산 프로그램(12종)
- * 비틀 변형과 응력 계산 프로그램 (30종)
- * 단면 성질(37종)
- * 강체 성질(18종)

여기서 소개하는 프로그램은 첫 번째로 기어 계산 (7종)중 표준 스트레이트 베벨 기어 계산식, 두 번째로 단위 환산 프로그램(12종)중 길이 단위환산 프로그램이다.

(1) 표준 스트레이트 베벨 기어 계산식

Fig. 5는 표준 스트레이트 베벨 기어 계산식에서 입력창의 모습이다.

입력

계산 항목	기호	값
기어 잇수	Z_1 (피니언) Z_2 (기어)	20 40
속각(degree)	S	90
모듈(mm)	m	3
입력각(degree)	α_0	20
치역(mm)	b	22

Fig. 5. 스트레이트 베벨 기어 계산식의 입력창

표준 스트레이트 베벨 기어 계산식에서 입력창의 HTML 소스 파일을 살펴보면 아래와 같은 소스를 볼 수 있다.

```

<tr><td width="171" rowspan="2">
<p>기어 잇수</p></td>
<td width="171"><p>z<sub>1</sub>(피니언)</p></td>
<td width="171"><p><input type="text" name="z1"
value="20" OnKeyUp="calgear(this.form)"></p></td></tr>

```

위 소스의 네 번째 줄을 보면 input 데이터의 형식은 텍스트 형식이고 이름은 z1, 초기값은 20 으로 설정되어 있다. OnKeyUp 이란 명령어는 값을 입력시키는 즉시 JavaScript 함수를 실행하여 출력하도록 하는 명령어이다. OnKeyUp="calgear(this.form)" 구문은 입력받은 값을 JavaScript calgear란 함수로 넘겨 실행하고 실행값을 출력값으로 넘겨준다.

표준 스트레이트 베벨 기어의 여러 가지 출력값들을 계산해 주는 JavaScript의 calgear란 함수를 살펴보면 크게 세 부분으로, 입력값의 매개변수 정의, 출력값의 계산식 그리고 출력값의 매개변수 정의로 나누어진다.

```
var S = object.S.value*3.1415926535/180;
var z1 = object.z1.value;
var z2 = object.z2.value;
var m = object.m.value;
var alp0 = object.alp0.value*3.1415926535/180;
var b = object.b.value;
```

위 구문이 JavaScript에서 입력값의 매개변수를 정의하는 부분이다. 매개변수 z1과 같이 간단히 매개변수만 정의 할 수 있고, 매개변수 S와 같이 매개변수에 연산자를 사용하여 계산식을 첨가할 수 있다.

Fig. 6은 표준 스트레이트 베벨 기어 계산식의 출력창의 모습이다.

출력

치말 높이 (mm)	h_k	<input type="text"/>
치형 높이 (mm)	h_f	<input type="text"/>
전치 높이 (mm)	h	<input type="text"/>
피치 원지름 (mm)	d_{01}	<input type="text"/>
	d_{02}	<input type="text"/>
피치 원추각 (degree)	δ_{01}	<input type="text"/>
	δ_{02}	<input type="text"/>

Fig. 6. 스트레이트 베벨 기어 계산식의 출력창

```
var hk =1.0* m;
object.hk.value = hk;
var hf =1.25*m;
object.hf.value = hf;
var h= 1.0*hk+1.0*hf;
object.h.value =h;
```

```
var d01 = m*z1;
object.d01.value =d01;
var d02 = m*z2;
object.d02.value =d02;
var del01 = atan(sin(S)/(z2/z1+cos(S)))*3.1415926535/180;
object.del01.value = del01;
var del02 = S - d01*3.1415926535/180;
object.del02.value = del02;
```

위 JavaScript 소스 구문은 HTML에서 넘겨받은 입력값을 계산하는 JavaScript의 계산식과 출력값의 매개변수를 정의하는 것이다. 각각의 계산식을 정의하고 그 출력값의 매개변수를 정의한다. 계산식에는 앞에서 정의된 입력값의 매개변수와 출력값에서 정의된 매개변수 모두 식에 포함되어 사용될 수 있다.

```
<tr> <td width="171"><p>치말 높이 (mm)</p></td>
<td width="171"><p>h<sub>k</sub></p></td>
<td width="171"><p><input type="text" name="hk" size="20"></p></td></tr>
```

출력창의 HTML 소스를 살펴보면 위와 같은 구문을 볼 수 있고, 출력창의 크기와 출력될 매개변수의 이름과 단위를 정의되어 있다. 또한 출력될 매개변수의 형식과 출력값 매개변수의 이름이 설정되어 있다. 여기서 설정된 이름은 JavaScript의 출력값의 매개변수에서 정의되어 있다.

```
with (Math) {
var del01 = atan(sin(S)/(z2/z1+cos(S)))*3.1415926535/180;
}
```

여러 가지 출력값들을 계산해 주는 JavaScript의 calgear란 함수를 살펴보면, 위와 같이 with (Math) { }란 명령어를 볼 수 있는데, 여기 중괄호 안에서는 수학적 함수를 자유롭게 사용할 수 있다. atan(\tan^{-1}) 뿐만 아니라 pow(거듭제곱), sin, cos, tan, π (PI) 등의 수학적 함수를 자유롭게 사용할 수 있다.

(2) 길이 단위환산

Fig. 7은 길이 단위환산에서 입출력창의 모습이

다. 아래는 길이 단위환산에서 입력창의 HTML 소스이다.

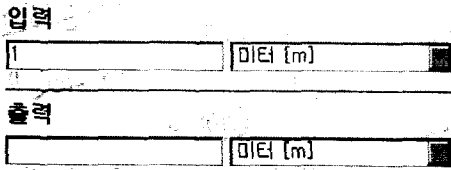


Fig. 7. 길이단위환산 입출력창

```
<tr><td colspan="2"><p><strong>입력</strong></td></tr>
<tr><td><p><input type="text" name="ValueFrom"
value="1" onkeyup="convert(this.form)"></td>
<td><p> <select name="UnitFrom" onchange
="convert(this.form)">
<option value="0.001">밀리미터 [mm]</option>
<option value="0.01">센티미터 [cm]</option>
<option value="0.3048">피트 [ft]</option>
<option value="0.0254">인치 [in]</option>
<option value="1000">킬로미터 [km]</option>
<option selected value="1">미터 [m]</option>
</select> </td></tr>
```

입력 데이터의 형식은 텍스트 형식이고 이름은 ValueFrom, 초기값은 1, OnKeyUp="convert(this.form)" 구문은 입력받은 값을 즉시 JavaScript convert란 함수로 넘겨 실행하고 실행값을 출력값으로 넘겨준다. 여기서 사용한 창 의 형식은 full down menu를 사용하였다. 그리고 처음에 보여지는 값이 미터[m]값이 설정 되도록 하였다.

아래는 길이 단위환산에서 출력창의 HTML 소스이다.

```
<tr><td colspan="2"><p><strong>출력</strong></td></tr>
<tr><td><p><input type="text" name="ValueTo" onkeyup
="convert(this.form) "></td><td><p>
<select name="UnitTo"onchange="convert(this .form)">
<option value="0.001">밀리미터 [mm]</option>
<option value="0.01">센티미터 [cm]</option>
<option value="0.3048">피트 [ft]</option>
<option value="0.0254">인치 [in]</option>
<option value="1000">킬로미터 [km]</option>
<option value="1">미터 [m]</option>
</select> </td></tr>
```

출력 데이터의 형식도 입력 데이터의 형식과 같

이 텍스트 형식이고 이름은 ValueTo, 초기값은 1, 역시 OnKeyUp="convert(this.form)" 구문을 사용하여 입력받은 값을 즉시 JavaScript calgear란 함수로 넘겨 실행하고 실행값을 출력값으로 넘겨준다. 여기서도 full down menu 형식을 사용하였다. 그리고 처음에 출력되어 보여지는 값 역시 미터 [m]값으로 설정 되도록 하였다.

다음은 길이 단위환산에서 출력값들을 계산해 주는 JavaScript의 convert란 함수를 살펴보면 크게 세 부분으로, 입력값의 매개변수 정의, 출력값의 매개변수 정의 그리고 출력값의 계산식으로 나누어진다.

```
<script language="JavaScript">
<!--
function convert(object) {
var UF = object.UnitFrom.value;
var UT = object.UnitTo.value;
var input = object.ValueFrom.value;
object.ValueTo.value = input*UF/UT;
} //--> </script>
```

4. 토의 및 결론

본 연구의 목적은 JavaScript를 이용한 웹 기반 공학계산 프로그램 개발이다. 이를 위해 JavaScript의 구동 원리를 소개하고 HTML 소스와 연동 방법을 자세히 설명하였다.

동국대학교 기계공학과에서는 그 동안 이와 비슷한 원격제어 시스템을 이용하여 함수발생기, LCR 미터, 오실로스코프를 컴퓨터에 연결하고 이를 제어하는 클라이언트 프로그램을 개발하였다. 또한 이들 프로그램들을 원격으로 구현하는 서버 프로그램을 작성하였는데 구축된 환경의 타당성 및 교육의 효과를 조사하기 위하여 다이오드 정류 실험, 트랜지스터 증폭 회로 실험, OP-AMP를 이용한 미분기 회로 실험을 수행하였다. 원격 실험 결과 측정 장비를 이용하는 공학 실험 실습 교육이 인터넷 상에서 충분히 구현 가능함을 알 수 있었으며 앞으로 좀 더 다양한 실험이 개발될 경우 효과적인 교육 매체로서 사용될 수 있을 것으로 예상된다.

앞으로 초기 연구 단계에서 확인된 원격 CCD

Camera 기술이 포함된다면 좀더 재미있고 흥미로운 원격 계측 실험실을 구축할 수 있을 것으로 기대된다. 본 연구에서 개발된 프로그램은 C++ 프로그램을 기반으로 하고 있으나 앞으로 JAVA 나 ASP 기반의 웹 프로그램이 활성화 될 경우 좀더 사용자에게 편리한 환경을 구축할 수 있을 것으로 예상된다.

5. 후기

본 논문은 중소기업청 산학연컨소시엄과제의 지원에 의하여 이루어졌다. 이에 관계자 여러분께 감사드린다.

참 고 문 헌

- [1] 김혜진, (한번 더 생각한) 자바스크립트, 대림, 서울, 1997

- [2] 광문규, 인터넷을 이용한 원격계측 시스템 개발, 한국소음진동공학회 2000년도 추계학술대회, 2000년 11월, pp.322-326

- [3] 이동준선생의 Java실험실 URL :
(<http://www.science.or.kr/lee/>)

- [4] 광문규, 조형제, 박준영, 이의수, 이호용 (1998) “가상실험 실습실 구현에 관한 연구”, 한국공학기술학회 공학교육학술대회, 1998년 11월, pp. 109-114.

- [5] 한상훈, 유성현, 조형제 (2000), “공학용 가상 실험실을 위한 Web용 기반 시스템의 구축”, 공학교육 연구, 제 3권 1호, 2000년 6월, pp. 27-41.

- [6] 교영숙, JavaScript 바이블, 영진, 서울, 1997