

Motor Drive 설계를 위한 Spice 용 Interface Tool 제작

이상용, 고재석, 목형수, 최규하, 최홍순*, 김덕근*
건국대학교 전기공학과 * (주) 코모텍

The Development of the Interface Tool for the Designing of Motor Drive Using Spice

S.Y. Lee, J.S. Gho, H.S. Mok, G.H. Choe, H.S. Choi, D.G. Kim
Dept. of Electrical Eng. , Konkuk Univ *KOMOTEK

ABSTRACT

The parameter through the motor designing program is used to predict motor response and design the motor drive circuits. The application programs such as "Spice" and "Saber" are often used for these. However, making the electrical model of motor for these simulation tool is uncomfortable and impossible for general users. Therefore, in this paper, we develop the "Spice" library generation program with the motor designing program "Motor Expert". This program will assist the user to make the motor library comfortably and correctly

로 설계 및 모터의 입력에 대한 응답을 보기 위해서 많이 사용되어지는데, 이러한 시뮬레이션툴에서 사용하기 위한 모터의 전기적 모델을 만들어내는 것은 매 설계 때마다 반복작업을 해야하므로 불편하며 일반사용자에게는 불가능하기까지 하다. 따라서 본 논문에서는 국내에서 제작된 설계 프로그램인 "Motor Expert"를 이용한 모터설계 시에 "Spice"를 위한 모델을 라이브러리로 자동 생성해주는 연계프로그램을 제작하여 사용자의 편의를 도모하고, 정확한 모터 라이브러리를 제작하여 [1][2] 적용할 수 있도록 하였으며 "PSpice"와 "Matlab"으로 만든 모터 모델의 동특성과 비교하여 [3] 모델의 정확성을 비교하였다.

1. 서론

컴퓨터의 발달로 인해 모터설계시 기존에 개발자의 지식에 의존했던 방식에서 벗어나 최근에는 각종 모터설계용 프로그램이 개발되어 산업현장에서 의 모터설계에 활용되고 있다.

모터설계 응용 프로그램의 종류에는 국외에서 제작된 "Maxwell" 이나 "PC-DCM" 과 같은 프로그램이 있고 국내에는 "Motor Expert" 같은 프로그램이 있으며 대부분 유한요소해석법(FEM)으로 파라미터들을 생성하고 이 파라미터들은 구동부 회로설계 및 모터의 응답을 예측하는데 이용된다.

이들 프로그램을 이용하여 모터를 설계한 후에 "Spice"나 "Saber"등의 응용프로그램이 구동부 회

2. Spice

"Spice"는 미국 Berkeley 대학에서 만든 시뮬레이션 프로그램으로 내부에 수치해석 모듈과 기본적인 전기회로 시뮬레이션에 필요한 수동소자, 스위치 소자, 각종전원의 라이브러리가 이미 준비되어 있어, 타 프로그램 코어로의 사용이 용이하고 상용 프로그램인 "PSpice", "IsSpice" 의 경우에도 내부 코어로 "Spice"를 사용하고 있다

즉 기본적인 전기회로의 경우에 "Spice" 코어만으로도 시뮬레이션이 충분하며 기타 공학분야에서는 내장된 기본모델 (Primitive Model)을 이용하여 필요한 모델을 예약 모델 (Macro Model)화 하여 시

물레이션 할 수 있다.

본 프로그램에서는 모터의 동특성을 규정 짓는 미적분 방정식들을 “Spice”가 코어인 “IsSpice”내의 Behavioral Model로 변환하여 모터 라이브러리를 제작하였다 [4]

3. 영구자석형 DC 모터의 모델

전기적 모델링과 비교해서 기계적 모델링을 할 때 유의할 점은 전기회로 시뮬레이션 프로그램 내에서 한 도선의 정보는 전압과 전류 두 가지 값만을 가지므로 기계적인 모델도 한 도선에서 두 가지 이상의 정보를 담을 수 없다는 제한이 생긴다. 따라서 전압, 전류의 관계와 유사한 형태로 출력을 결정하고 두 가지 이상의 정보의 출력은 하나 이상의 노드를 출력으로 가지게 하는 수밖에 없다. 영구자석형 DC 모터의 주요한 파라미터들을 모델링한 그림은 아래와 같이 나타낼 수 있다

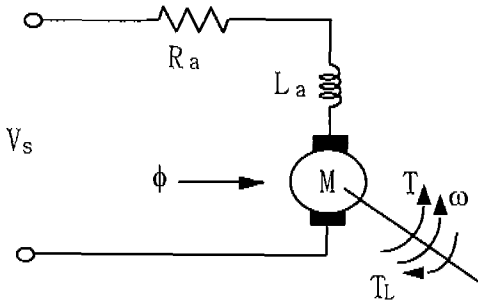


그림 1. 영구자석형 DC 모터의 모델

여기에서 $R_a[\Omega]$ 는 아마추어의 총 저항을 나타내고, $L_a[H]$ 는 아마추어의 총 인덕턴스이며, $V_s[V]$ 는 인가전압, $T[N \cdot m]$ 와 $T_L[N \cdot m]$ 은 각각 모터의 발생 토크와 부하의 토크를 의미하며 $\omega[rad]$ 는 회전자의 각속도, $\Phi[Wb]$ 는 공극의 총 Flux로서 영구자석에서의 발생 양이다.

영구자석형 DC 모터를 수식적으로 크게 두 가지로 나누어보면 아마추어의 전기적인 수식과 모터 회전자와 물리적 부하간의 기계적 수식으로 나눌 수 있다

아마추어의 모든 저항과 인덕턴스를 R_a, L_a 에 집중되었다고 보면 인가 전압 V_s 와 아마추어 전

류 I_a 의 전기적인 수식은 아래 식(1)과 같다.

$$V_s = R_a I_a + \frac{di_a}{dt} L_a \quad (1)$$

식(1)을 Laplace 변환하고 전류에 대한 표현으로 고치면 식(2), (3)과 같다

$$V_s(s) = (R_a + sL_a)I_a(s) \quad (2)$$

$$I_a(s) = \frac{V_s(s)}{(R_a + sL_a)} \quad (3)$$

모터 회전 시에 역기전력(E_g)의 발생으로 인해 식(2), (3)은 아래 식(4), (5)와 같이 표현된다.

$$V_s(s) - E_g(s) = (R_a + sL_a)I_a(s) \quad (4)$$

$$I_a(s) = \frac{V_s(s) - E_g(s)}{(R_a + sL_a)} \quad (5)$$

역기전력 E_g 는 식(6)과 같이 표현된다.

$$E_g = K_e \Phi \omega \quad (6)$$

(K_e : 역기전력 상수, ω : 회전자의 회전각속도)

또한 토크는 회전자 전류에 비례하며 식(7)과 같다.

$$T = K_t \Phi I_a \quad (K_t : \text{토크 상수}) \quad (7)$$

모터의 발생 토크는 Damping을 무시하면 식(8)과 같이 발생 토크와 부하 토크, 모터의 관성, 회전자 속도와의 관계식을 만족한다.

$$T = J \frac{d\omega}{dt} + T_L \quad (8)$$

(T : 모터 발생 토크, J : 모터의 관성,

T_L : 부하 토크)

마지막으로 자속은 실제로는 히스테리시스 특성 곡선을 따라 비선형적으로 추종하지만 이를 선형화 한 상수의 도입으로 모델링할 수 있다.

이상의 모터방정식에 기초한 모델의 블록 다이어그램은 아래 그림 2와 같다.

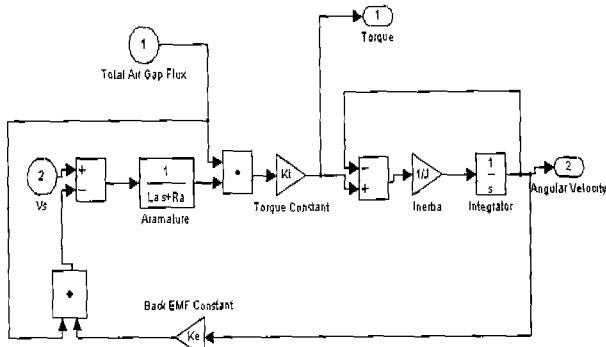


그림 2. 영구자석형 DC 모터의 블록 다이어그램

블록 다이어그램에 기초하여 상용 시뮬레이션 프로그램인 "IsSpice"에서 모델링한 영구자석형 DC 모터의 모델은 아래 그림 3과 같다

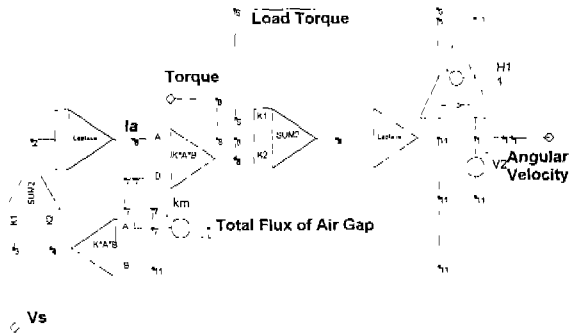


그림 3. 영구자석형 DC 모터의 IsSpice용 모델

출력단은 식(8)에서 각속도(ω) 대신 전압을 토크 대신 전류를 대입한 방정식을 모델화 하였으며 부하에 의존하는 출력의 전류 값이 토크를 의미한다. 따라서 물리적인 부하도 적당한 치환을 통해 수동 소자들로 모델링 할 수 있다.

4. 프로그램의 구성

본 논문의 프로그램은 국내에서 제작된 모터설계 프로그램인 "Motor Expert"를 대상으로 만들어진 프로그램으로서 단독으로 파라미터 입력을 받고 자동으로 라이브러리를 생성하는 동작과 파라미터를 기존 모터설계 프로그램에서 넘겨받아 라이브러리를 생성하는 동작을 할 수 있게 설계되었다.

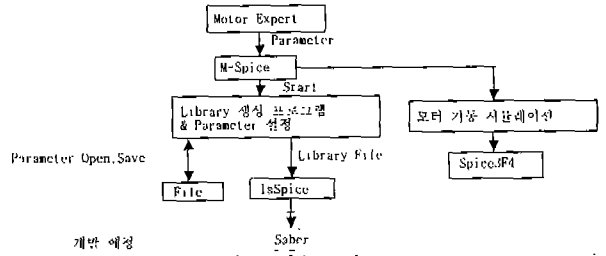


그림 4. 프로그램 구성도

"M-Spice"로 가명한 본 프로그램은 "Motor Expert"로부터 필요한 파라미터를 모두 넘겨받아 자동으로 모델을 생성할 수도 있고 사용자가 직접 모든 파라미터를 설정할 수도 있다.

본 프로그램의 입력 파라미터 설정 및 확인용 대화박스는 아래와 같다.

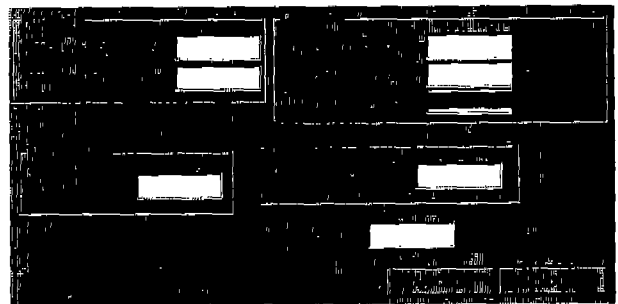


그림 5. 입력 파라미터 설정 및 확인용 대화박스

또한 출력 모델을 보기 위한 대화박스는 아래와 같다.

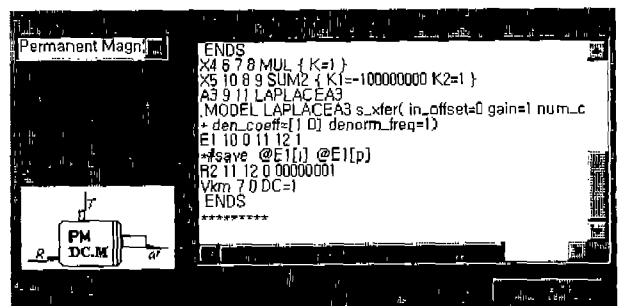


그림 6. 출력 결과 확인용 대화박스

5. 시뮬레이션 결과

본 프로그램으로 만든 영구자석형 DC 모터의 라이브러리로 "IsSpice" 내에서 시뮬레이션 하였다.

시뮬레이션조건은 $R_a = 1.46 [\Omega]$, $L_a = 25m [H]$,
 이었으며 정격전압은 220[V], 모터의 관성은 0.15
 $[K_g m sec^2]$ 이고 $K_e = K_t = 1$ 로 가정하였으
 며 영구자석에서 발생하는 총자속은 0.8[Wb] 이다.
 본 프로그램을 통해 자동 생성된 라이브러리의 코
 드는 아래와 같다.

```
.SUBCKT pmdc 3 12 8
*FAMILY = ANALOG
*FAMILY = ANALOG
*FAMILY = ANALOG
H1 5 0 v2 1
*#save @H1[i] @H1[p]
A2 2 6 LAPLACEA2
.MODEL LAPLACEA2 s_xfer( in_offset=0 gain=1
num_coeff=[1]
+ den_coeff=[25m 1.46] denorm_freq=1)
X1 7 11 4 MUL { K=1 }
.SUBCKT MUL 1 2 3 {K=???}
B1 3 0 V = V(1) * V(2) * {K}
.ENDS
V2 1 11
X3 3 4 2 SUM2 { K1=1 K2=-1 }
.SUBCKT SUM2 1 2 3 {K1=??? K2=???}
B1 3 0 V = {K1}*V(1) + {K2}*V(2)
.ENDS
X4 6 7 8 MUL { K=1 }
X5 5 8 9 SUM2 { K1=-1 K2=1 }
A3 9 11 LAPLACEA3
.MODEL LAPLACEA3 s_xfer( in_offset=0 gain=1
num_coeff=[6.7]
+ den_coeff=[1 0] denorm_freq=1)
Vkm 7 0 DC=0.8
.ENDS
```

결과로 "IsSpice" 내에서 시뮬레이션한 출력 각속
 도는 그림 7과 같고 출력 Torque는 그림 8과 같다
 "Pspice" 내에서 동일한 모델을 시뮬레이션 했을
 때의 출력은 그림 9와 같고 "Matlab Simulink"에
 서 시뮬레이션한 출력 각속도와 토크는 그림 10과
 같다.

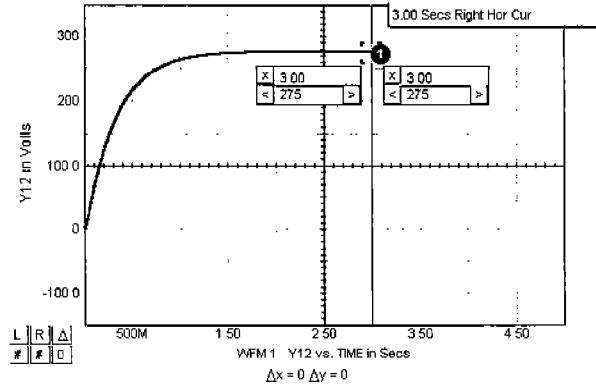


그림 7. IsSpice 시뮬레이션 출력 (각속도)

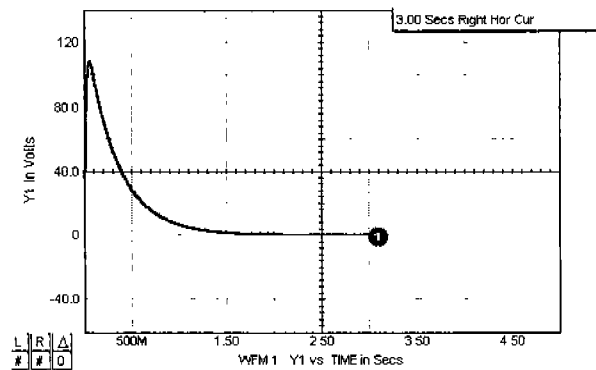


그림 8. IsSpice 시뮬레이션 출력 (토크)

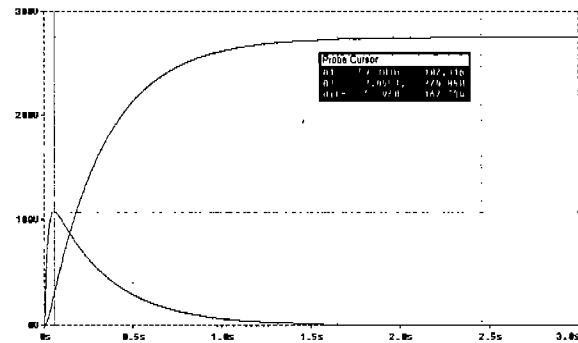


그림 9. PSpice 시뮬레이션 출력 (각속도, 토크)

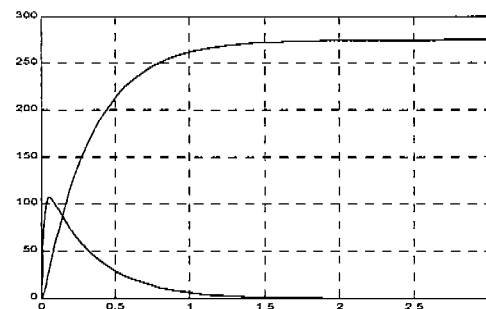


그림 10. Matlab 시뮬레이션 출력 (각속도,토크)

"IsSpice", "PSpice", "Simulink"에서의 시뮬레이션 출력을 비교해볼 때 각 프로그램이 연산방식의 차이와 사용한 기초모델의 차이가 있었지만, 결과를 비교해볼 때 동일한 동태를 보인다.

즉 시뮬레이션 결과를 통하여 모델의 정확성을 상대적으로 비교할 수 있었고, 본 프로그램을 통해 영구자석형 DC 모터의 모델을 만드는데 걸린 시간은 불과 수분이었다.

6. 결론

본 논문에서는 사용자입장에서 편리하고, 정확한 모터 라이브러리를 생성하는 프로그램을 작성하였다.

파라미터 값들은 모터설계 툴인 "Motor Expert"를 통해서 받았고 "IsSpice"용 라이브러리를 자동으로 생성하였으며 시뮬레이션을 통하여, 그 출력 값을 타 시뮬레이션 프로그램 결과와 비교하여 보았다. 향후에 BLDC 모터, 유도기, 동기기 등 자동 작성할 수 있는 모델의 확장으로 본 프로그램의 유용성을 더욱 높일 수 있을 것이라고 생각하며 모터 뿐만 아니라 반도체소자 등 다른 요소들에 대해서도 자동으로 시뮬레이션 프로그램과의 인터페이스용 라이브러리를 생성하는 프로그램을 만든다면 사용자 입장에서 유용한 프로그램이 될 것이라고 생각된다.

참고문헌

- [1] Lawrence G.Meares, Charles E.Hymowitz , "Spice Applications Handbook", Intusoft, pp 19-1~20-10, 1990
- [2] 박준기,백정렬 , "Microsoft Visual C++ MFC Library Reference , 1998
- [3] Lawrence G.Meares, Charles E.Hymowitz , "Simulation with Spice", Intusoft, pp1-2~3-13, 1988
- [4] H.Alan Mantooth, Mike Fiegebaum , "Modeling with an Analog Hardware Description Language", Analogy Inc. , pp.13~64, 1995