

Simulink/RTW를 이용한 실시간 제어시스템 개발

강 문호, 정 대연
선문대학교 공과대학 기계 및 제어공학부

Development of a Real Time Control System Using Simulink/RTW

Kang, Moon-Ho, Jeong, Dae-Yeon
Div. of Mechanical and Automatic Control Eng., Sun Moon Univ.

Abstract - In this research a real time control system was developed without program codings during control system designing procedures. On the Simulink window control system is designed in the form of block diagrams, program codes are produced automatically with the real time workshop package(RTW), then c compiler compiles the program codes. With this automatic real time program producing mechanism rapid prototyping is realized. To show effectiveness of the proposed designing scheme a DSP-based induction motor vector control system was constructed and implemented

1. 서 론

제어이론과 제어기 설계기술이 발전함에 따라 제어 시스템이 더욱 복잡 다양화되어 미분 방정식이나 행렬식 등을 이용하여 제어시스템을 일일이 수동으로 해석 및 설계하는 것은 매우 어렵고 많은 시간이 소요되는 경우가 많다. 또한 해석방법에 있어서도 복잡한 계산 뿐만 아니라 다양한 그래프를 포함하는 사용자 인터페이스작업을 동반하는 경우도 많아 제어시스템 설계 과정에서 필수적으로 동일한 일의 반복작업이 포함되는 특성을 가진다. 따라서 이러한 문제를 해결하는 데 있어서 컴퓨터의 사용이 필수적으로 되어있다. 그러나 이러한 문제점들을 해결하기 위해 종래의 방법인 Fortran, C, C++ 등을 이용하여 직접 프로그래밍 하는 방식은 제어부설계뿐만 아니라 그래프출력, 데이터의 입·출력, 컴퓨터와 제어기간의 통신을 담당하는 프로그래밍 등 부가적인 일에도 많은 시간을 들여야 하므로 용이한 작업이 아니다. 본 연구에서는 이러한 문제점을 해결하기 위해서 제어시스템 설계에서 실행파일 생성 및 구현까지의 전과정이 프로그램 코딩없이 진행되는 실시간 제어시스템을 개발하였다. Simulink[1] 윈도우상에서 Simulink 블록들과 S-Function[2]으로 설계된 사용자생성블록들을 이용하여 블록 다이어그램 형태로 제어시스템을 설계하고, Real Time Workshop (RTW)[3]을 통해 제어시스템 구현을 위한 C코드를 자동 생성시킨 후, C컴파일러를 이용하여 컴파일과 링크를 수행하고 실행파일을 생성한다. 생성된 실행파일은 PC의 병렬포트를 통해 DSP(TMS320C32)[4]보드로 다운로드 되어져 제어작업을 수행한다. 한편, 제어작업중 발생하는 데이터들은 Matlab의 mat파일[3]형태로 변환되어 병렬포트를 통해 PC로 업로드되어져 Matlab그래프상에서 확인될 수 있도록 하였다. 개발된 시스템의 유용성을 보기 위하여 DSP와 IPM[5]을 이용한 유도전동기 벡터제어[6] 장치를 제작하고 Simulink 윈도우상에서 제어시스템을 설계한 후, 실험을 행하여 시스템 성능을 고찰하였다.

2. 시스템 구성

2.1 소프트웨어 구성

제어시스템의 rapid prototyping을 위한 가장 중요한 요소는 시스템 설계시 필요한 제어알고리즘, 제어결과 분석을 위한 데이터의 처리 및 호스트(컴퓨터)와 플랫폼(DSP)사이의 통신 등을 구현하기 위한 프로그램 코드를 자동으로 생성, 컴파일, 링크하므로써 기존에 수작업에 의한 처리와 비교하여 시스템 개발에 소요되는 시간을 대폭 줄이는 기능이다. 본 연구에서는 Math Work사의 Simulink와 Real Time Workshop(RTW)을 이용하여 시스템 설계시 필요한 코드들을 자동생성한 후, 생성된 C코드들과 TI사의 C-컴파일러를 연계시켜 컴파일, 링크시키고 DSP에서 구동되는 실행파일을 생성한다. 이러한 일련의 과정을 단계별로 설명하면 다음과 같다.

①시뮬링크윈도우상에서 제어시스템구성(ex.test.mdl). 이때, 시스템을 구성하는 제어알고리즘, 사용자가 작성한 I/O인터페이스부, 인터럽트처리부 및 모듈실행부, 실험데이터 입/출력부 등을 각각 블록형태로 시뮬링크 윈도 윈도우상에서 설계.

<RTW의 System Build과정>[3]

- ②RTW의 make_rtww.m 파일이 test_rtww파일(test.mdl이 지니는 모든 블록정보를 가지는 파일)을 생성
- ③RTW의 make_rtww.m 파일이 Target system(ex. DSP320C32)에 대응하는 특정한 TLC(Target Language Compiler, ex. grt.tlc)를 호출하고 TLC는 test_rtww로부터 블록정보들을 수집하여 test.c, test.reg, test.prm, test.h등의 ANSI C코드들 생성
- ④make_rtww.m 파일이 rtww_c.m 파일을 호출하여 사용자가 작성한 templatemake 파일(ex. test.tmf)로부터 make 파일인 test.mk파일을 생성
- ⑤make_rtww.m 파일이 VC++가 제공하는 nmake.exe를 호출하여 test.mk에 정의되어 있는 순서대로 make작업을 수행.

<nmake의 실행파일 생성과정>

- ⑥nmake.exe는 test.mk에 정의되어 있는 순서대로
- ③번 과정에서 생성된 test.c, test.reg, test.prm, test.h들, RTW에서 제공하는 라이브러리 C코드들(ex. rt_hypot.c, rt_look.c, rt_matrix.c, rt_sgn.c, rt_zcfnc.c, rt_rand.c), 사용자가 작성한 S-function C코드 및 기타 C코드들을 컴파일, 어셈블, 링크시킨 후, 실행파일(test.out)생성. 이때, nmake.exe는 DSP 컴파일러가 제공하는 응용프로그램들을 호출해 가면서 컴파일, 어셈블, 링크, 실행파일(test.out)생성등을 수행
- ⑦최종적으로 nmake.exe는 실행파일(test.out)로부터 DSP보드에 다운로드 가능한 파일포맷을 갖는 hex파일(test.hex)을 생성

〈실행파일 구현〉

- ⑧ VC++로 작성된 응용파일들중 다운로드파일(ex. send.exe)을 시물링크 윈도우상에서 호출하고, 다운로드파일이 hex사파일(test.hex)을 PC로부터 DSP로 직렬 전송
- ⑨ VC++로 작성된 응용파일들중 프로그램수행파일(ex. play.exe)을 시물링크 윈도우상에서 호출하고, 프로그램수행파일은 DSP에 직렬전송된 hex사파일(test.hex)을 수행.
- ⑩ 필요한 경우 VC++로 작성된 응용파일들중 데이터전송파일(ex. receive.exe)을 시물링크 윈도우상에서 호출하고, 데이터전송파일은 DSP로부터 PC로 실험데이터를 직렬전송시켜 실시간으로 실험데이터를 분석가능하게 하거나, 데이터파일의 생성, 저장, 가공이 가능하게 함.

상기한 일련의 과정을 보이면 Fig.1과 같다.

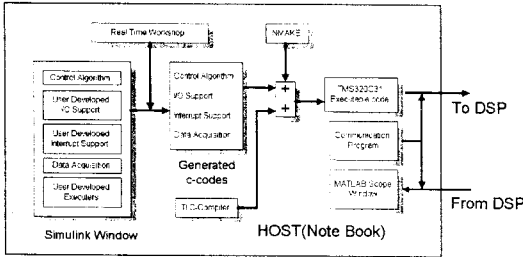


Fig.1 Automatic code generation, compile and link operations flow.

2.2 하드웨어 구성

실험에 사용된 하드웨어의 구성을 보이면 Fig. 2와 같다. Simulink, RTW와 C-컴파일러를 비롯한 시스템 설계에 필요한 프로그램들은 노트북에 탑재된다. RTW에 의해 C-코드들을 생성한 후 이를 컴파일, 링크시킨 후 발생되는 실행 이진 파일은 노트북의 프린터 포트를 통해 ADC, 카운터, 인버터구동을 위한 공간벡터 PWM발생부등의 I/O와 메모리, 각종 보호회로와 주변 장치 등을 갖는 DSP보드로 다운로드 된다.

DSP는 제어알고리즘에 따라 제어입력을 계산하고 이 값은 공간벡터PWM방식[7]에 의해 전압변조되어 IPM을 채용한 인버터에 인가되어 유도전동기를 구동한다.

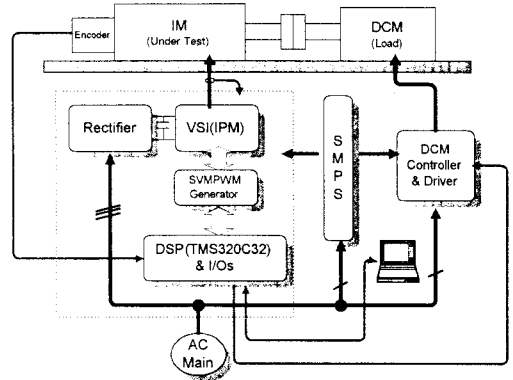


Fig. 2 Hardware configuration.

3. 실험

개발된 시스템의 유용성을 확인하기 위해서 제작된 유도전동기 벡터제어 장치에 디지털 비례-적분제어기법을 적용하여 전동기 속도 및 비간섭 전류제어[6]를 행한 후 각각의 제어성능을 비교하였다. 인버터구동을 위한 전압변조는 공간벡터PWM방식을 사용하였고 8254타이머를 이용하여 구현된다. 이를 위한 소프트웨어부분이 Fig.3의 8254 pattern generation S-function 블록에서 작성되었다. 비간섭전류제어를 위한 디커플링부는 Fig.3의 decoupling controller블록에서 작성되었다. Simulink 윈도우상에서 구성된 속도 PI제어기를 보이면 Fig.4와 같고, 적분제어기의 경우 제어기 출력포화에 따른 과도현상을 없애기 위하여 적분기 windup을 방지하는 기능을 첨가하였다. 전류제어기의 경우도 속도 제어기와 같은 구조를 갖도록 선택하였고, 속도 및 전류 제어기의 제어이득들은 Table1의 전동기 파라미터에 근거하여 Table2와 같이 설정하였다. Fig.3은 Simulink 윈도우상에서 구성된 전체제어시스템 블록도이다.

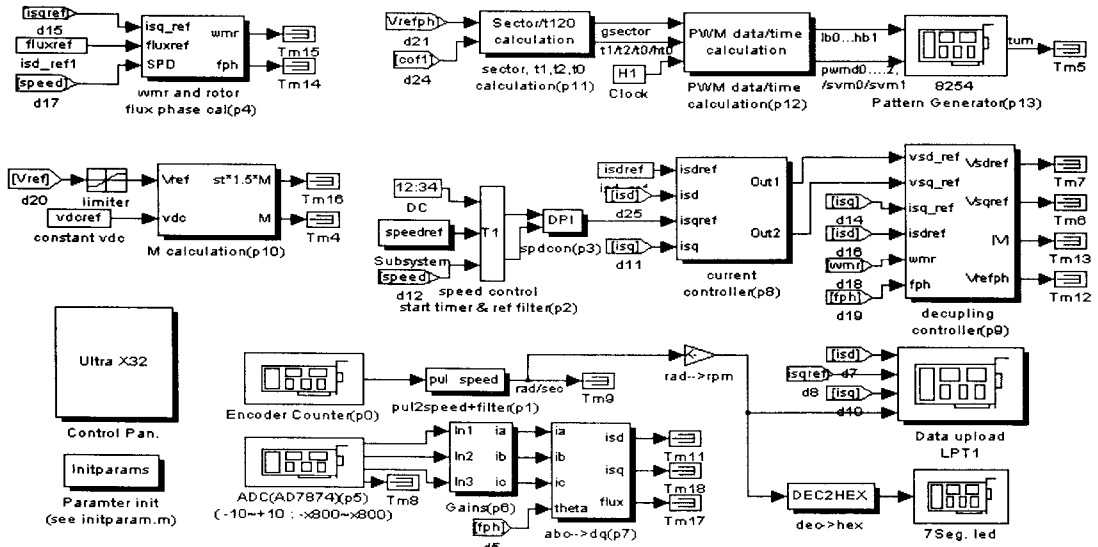


Fig. 3 Simulink model for an induction motor vector control system.

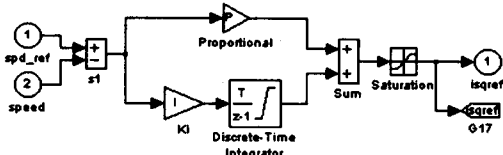


Fig.4 Induction motor PI speed controller. (Internal configuration of the DPI block in Fig.3)

Fig. 5는 속도를 1500rpm에서 500rpm으로 가감속시킨 경우 유도전동기 벡터제어시 응답특성들을 보인다. Fig. 5에서 (a)는 자속성분전류(isd) 응답특성을 보이고, (b)는 토크성분전류의 기준값(isqref)이고, (c)는 토크성분전류응답이다. (d)는 속도응답특성이다. 자속성분전류의 기준값은 3[A]로 설정하였다. 실험결과, 속도 및 전류들이 전동작영역에 걸쳐 기준값을 잘 추종하고 있음을 알 수 있다. 실험결과 데이터들은 실험 종료후 DSP보드로부터 PC로 Matlab의 mat파일 형태로 업로드된 후 Matlab의 plot함수를 이용하여 도시되었다. Fig. 6은 감속시 제어특성을 자세히 관찰하기 위해서 감속기간의 응답을 확대하여 도시한 것이다.

Table 1. Parameters of Induction Motor.

3 ϕ , 4P, 220V, 60Hz, 3.6A	
Rs	3.63 [Ω]
Rr	3.87 [Ω]
Ls	0.104 [H]
Lr	0.103 [H]
Lm	0.1 [H]
J	0.012 [Kg m ²]

Table 2. Controller gains.

제어주기	속도제어	1ms
	전류제어	200 μ s
속도제어이득	Kp:0.3 Ki:62.4	
전류제어이득	토크성분전류제어	Kp:30 Ki:3120
	자속성분전류제어	Kp:30 Ki:3120

3. 결 론

본 연구에서는 제어시스템 설계시 프로그램 코딩이 필요하지 않은 실시간 제어시스템을 개발하였다. Simulink 윈도우상에서 블록 다이어그램 형태로 제어시스템이 설계되고, 설계된 제어시스템 구현을 위한 C코드의 생성 및 컴파일과 링크가 자동으로 이루어져 실시간 제어시스템 설계시 설계기간을 단축하고 신뢰성을 높일수 있는 rapid prototyping을 위한 실시간 제어시스템 설계 환경을 구축하고 DSP와 IPM으로 구성되는 유도전동기 벡터제어 드라이브를 제작하였다. 제시된 시스템 설계방식의 유용성을 보이기 위해, 속도제어와 전류제어 방식으로 디지털 PI제어기법을 선택하여 Simulink 윈도우상에서 유도전동기 벡터제어 시스템을 설계한 후 실험을 통해 제어성능을 분석하였다.

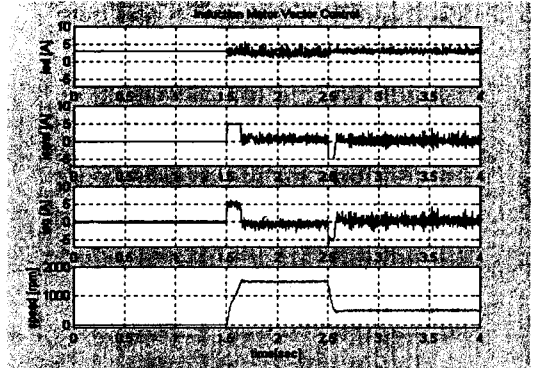


Fig. 5 Induction motor vector control response. ((a)flux current(reference:3(A)) (b)torque current reference (c)torque current (d)speed)

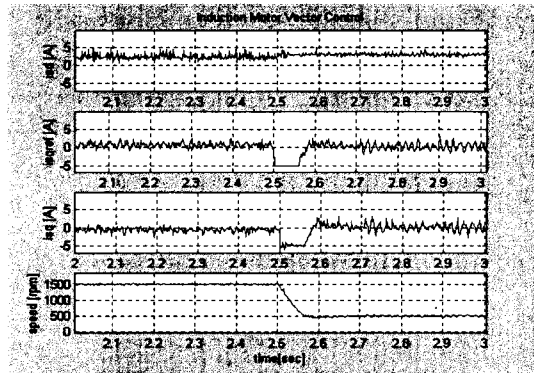


Fig. 6 Induction motor vector control response.(zoomed) ((a)flux current (b)torque current reference (c)torque current (d)speed)

감사의 글

본 연구는 한국과학재단 지정 전문대학교 공조기술 지역 연구센터의 지원에 의한 것이며 이에 감사를 드립니다

(참 고 문 헌)

- [1] Simulink, *Dynamic System Simulation for MATLAB, Using Simulink*, the Math Works Inc., 1998.
- [2] Simulink, *Dynamic System Simulation for MATLAB, Writing S-Functions*, the Math Works Inc., 1998.
- [3] *Real-Time Workshop, for Use with Simulink, User's Guide*, the Math Works Inc., 1998.
- [4] *TMS320C3X/4X Optimizing C Compiler User's Guide*, the Texas Instruments Inc., 1998.
- [5] E.R.Motto et al., "A New Generation of Intelligent Power Device for Motor Drive Applications", *IEEE IPEC*, pp.1332-1338, 1993.
- [6] P.Vas, *Vector Control of AC Machines*, Clarendon Press, Oxford, 1992.
- [7] H.W.Van der Broeck et al., "Analysis and Realization of a Pulse With Modulator Based on Voltage Space Vectors", *IEEE Trans. Ind. Appl.*, vol.24, no.1, pp.142-150, 1988.