

<http://dx.doi.org/10.7236/JIIBC.2013.13.2.9>

JIIBC 2013-2-2

## 안전성을 고려한 모델 기반 제어기 설계

### Design of a Model Based Controller with Safety

신범식\*, 박정훈\*, 문찬우\*\*, 안현식\*\*

Bum-Sik Shin, Jeong-Hoon Park, Chan-Woo Moon, Hyun-Sik Ahn

**요 약** 모델 기반 설계는 제품의 안전성을 확보하고 제품의 개발기간을 단축시키는 장점이 있다. 이 논문에서는 Simulink를 사용하여 BLDC 모터제어기의 모델 기반 설계를 수행한 후 자동 코드 생성방법에 의해 C코드를 생성하여 제어기를 구성하였다. 시스템의 안전성확보를 위하여 MISRA AC SLSF의 가이드에 따라 모델을 구성을 하였으며 제작된 실제 제어기를 사용하여 생성된 코드를 검증하였으며 각 하부 제어기의 수행시간을 측정하여 기존의 설계 방법과 모델기반 제어기의 설계 방법으로 구성된 제어기를 비교하였다.

**Abstract** Model based design method reduces product development period and increases system software safety. In this paper, a BLDC motor controller based on model based design method is designed with Simulink and implemented with auto generated code which is written in C language. To retain the safety of software, this model is implemented according to MISRA AC SLSF guide. The validity of the implemented controller is verified with a real position control experiment, and execution times of each control loops are measured to compare the system performance of the conventional design and the model based design.

**Key Words :** Model based design, BLDC motor controller, Auto code generation, MISRA AC SLSF, Safety

## 1. 서 론

디지털 방식의 제어장치를 구성할 때 기존에는 C언어나 C++과 같은 프로그램 언어를 사용하여 하드웨어에 특화된 소프트웨어를 작성하는 방법으로 제어장치의 알고리즘을 구현하였다. 이 방법은 프로그래머가 알고리즘을 프로그램 언어로 변환하여야하며 또 하드웨어가 변경될 때마다 프로그램을 같이 변경하여야 하고, 하드웨어가 완성될 때까지는 알고리즘의 검증이 어려운 문제가 있었다. 또한 프로그램에는 필연적으로 오류가 존재하게 되어 시스템의 오동작을 발생시키는 원인

이 되므로 높은 안전도가 필요한 시스템에서는 새로운 디자인 방법이 요구되고 있다. 모델기반 제어기는 수학적 모델을 사용하여 MIL (model in the loop)을 구성하고 검증한 후 코드를 생성하여 제어기를 구성한다. 모델기반 설계방법에서는 알고리즘의 검증이 하드웨어와는 독립적으로 이루어질 수 있다. 모델 기반 방법에서는 안전성 문제와 최적화 문제가 주로 다루어진다<sup>[1][2]</sup>. 안전성은 프로그램의 오류에 의한 오동작에 의해 위해가 발생하지 않는 것을 의미한다. 모델 기반 설계를 하는 과정에서, 생성된 프로그램이 정해진 시간 주기 안에 수행되는 것이 보장되지는 않는데 이때는 프로그램의 크

\*준회원, 국민대학교 전자공학부

\*\*정회원, 국민대학교 전자공학부

접수일자 2013년 2월 15일, 수정완료 2013년 3월 15일

게재확정일자 2013년 4월 12일

Received: 15 February 2013 / Revised: 15 March 2013 /

Accepted: 12 April 2013

\*\*Corresponding Author: mcwnt@kookmin.ac.kr

School of electrical engineering, Kookmin University, Korea

기나 수행시간의 면에서 시스템의 최적화가 요구되는데, 많은 시스템에서 안전성과 최적화는 서로 대립되는 요소가 된다. 일부 시스템, 특히 차량용 시스템인 경우 높은 수준의 안전성이 요구되는데, 기존의 설계 방법에서는 시스템 프로그램을 C언어로 작성 시 안전성 확보를 위해 MISRA(Motor Industry Software Reliability Association)에서 발간한 지침인 MISRA-C 가이드가 있으며<sup>[3]</sup>, Simulink에서도 모델로부터 자동 생성된 코드가 MISRA-C 가이드를 준수하도록 하는 디자인 기능이 제공되고 있다. 알고리즘의 테스트나 검증 등은 생성된 코드에서 보다는 모델 자체에서 수행하도록 권고되고 있기 때문에 모델 구현 단계에서의 안전성을 확보하도록 해야 한다. MISRA에서는 MISRA AC SLSF에서 Simulink 모델 구현에 대한 가이드를 제공하고 있다<sup>[4]</sup>. 이 논문에서는 모델기반 구현방법에 따라 BLDC 모터 제어기를 구현하며<sup>[5][6]</sup> 이때 MISRA AC SLSF 가이드에 따라 모델 구현을 진행한다. 모델에서 자동 생성된 코드는 마이크로컨트롤러에 탑재하여 검증하며, 자동 생성된 코드에 의한 제어기와 직접 코딩한 제어기의 작업수행시간을 비교한다. 본 논문의 구성은 다음과 같다. II장에서는 BLDC 제어의 원리와 세부 모델에 관해 서술하고 III장에서는 모델의 구현에 대해, IV장에서는 실험 결과를 기술하고 마지막 V장에서 결론을 맺는다.

## II. 제어 시스템의 구성

### 1. BLDC 모터 제어

BLDC 모터는 기동토크가 크고, 정속에 안정적이며 브러시가 없어 불꽃, 노이즈, 잡음 없이 고속운전이 가능하다는 특징 때문에 가전, 자동차, 항공, 우주, 의료, 자동화 장비 등과 같은 산업 분야에 넓게 사용되고 있다<sup>[6]</sup>. 전체 제어기의 구성은 그림 1과 같다. 위치제어기 속도제어기, 전류제어기가 직렬로 연결되어 있으며 모터에는 위치센서가 장착되어 있어 위치제어기와 속도제어기에 피드백되고 모터에 흐르는 전류는 전류센서에 의해 검출되어 전류제어기에 피드백된다. 각 제어기는 PID 제어기를 사용하며 내부의 전류제어기의 경우 제어기의 출력에 비례하는 듀티를 갖는 PWM 신호가 모터에 인가된다. 이 때 인가하는 방식으로 통상 2상 여자 방식을 사용하는 경우도 많지만, 정밀한 제어가 필요

한 경우 d-q상변환에 의한 벡터제어를 사용한다. BLDC 모터를 제어하기위한 하드웨어 구성은 그림 2와 같다.

### 2. 모델기반 시스템 디자인 절차

모델 기반 시스템 디자인은 MIL (Model-In-the-Loop), SIL (Software-In-the-Loop), PIL (Processor-In-the-Loop), RP (Rapid Prototyping)등이 있으며, 그림 3과 같이 개발 단계의 V 모델로 연결되어 있다<sup>[7][10]</sup>. MIL은 플랜트나 제어 알고리즘을 시뮬레이션을 통해 검증할 수 있는 모델로 제어장치의 게인 설정이나 강인성을 확인할 수 있는 장점이 있다. MIL은 그림 4와 같이 나타낼 수 있다.

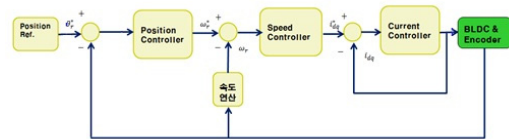


그림 1. BLDC 모터 제어 알고리즘 블록도  
Fig. 1. Block diagram of BLDC control

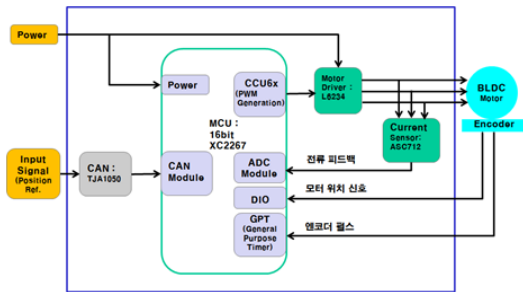


그림 2. BLDC 모터 제어 하드웨어 블록도  
Fig. 2. Block diagram of BLDC control hardware

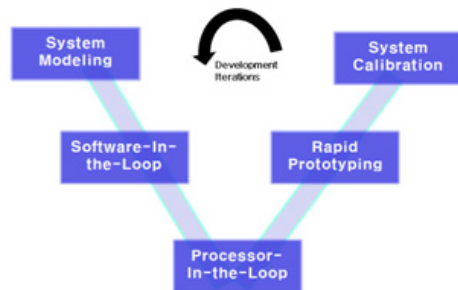


그림 3. 모델 기반 개발 방법의 V 모델  
Fig. 3. V model of model based design

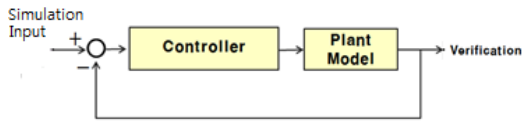


그림 4. MIL의 블록도  
Fig. 4. Block diagram of MIL

### III. 모델 구성

#### 1. MISRA AC SLSF 가이드에 의한 모델 구성

MISRA AC SLSF는 Mathworks의 Simulink와 StateFlow에 관한 55개의 가이드로 구성되어 있으며 각 가이드는 Required, Advisory, Document 중 하나의 성격을 갖는다<sup>[4]</sup>. 이 논문에서 제어장치를 구현할 때 사용한 대표적인 가이드는 다음과 같다.

##### 가. 데이터 타입

데이터 타입 변환이 필요한 부분에는 “Data type conversion block”만을 사용하여야 한다.

##### 나. Solver Setting

실제 제어장치에서 구현될 알고리즘에는 “Fixed step discrete solver”를 사용하여야 한다.

##### 다. 사용가능한 Simulink 블록

- 표준 Simulink 블록이 아닌 경우 코드 생성이 가능해야 한다.
- Function block, matlab 함수 block, 복제된 입력 포트 등은 사용하지 않는다.

허용되는 블록으로 구성된 PID제어기의 예 그림 5와 같다.



그림 5. 허용되는 블록을 사용한 제어기  
Fig. 5. Allowable Simulink Block

##### 라. Block Parameter

- 수행 중에 값이 결정되는 수식 블록 파라미터는 표

현, 데이터 타입 변환, 행이나 열의 선택 함수를 사용하지 않는다.

- 상수의 이름은 외부파일에서 선언한다.
- 상수값에는 상수이름을 지정한다.

##### 마. 초기화

초기값이 필요한 블록은 명확하게 초기값을 설정하여야 한다.

##### 바. 산술연산의 순서

덧셈 블록의 경우 첫 번째 입력은 “+”로 하고 곱셈 블록에서는 첫 번째 입력을 “\*”로 설정한다. 연산의 순서를 준수한 덧셈블록으로 표현한 PID 블록은 그림 6과 같다.

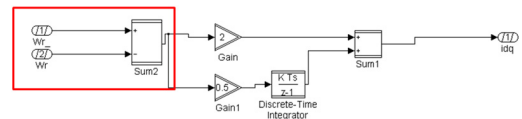


그림 6. 산술 연산의 순서  
Fig. 6. Arithmetic operation order

시뮬레이션을 위한 플랜트를 포함한 전체 모델은 그림 7과 같다.

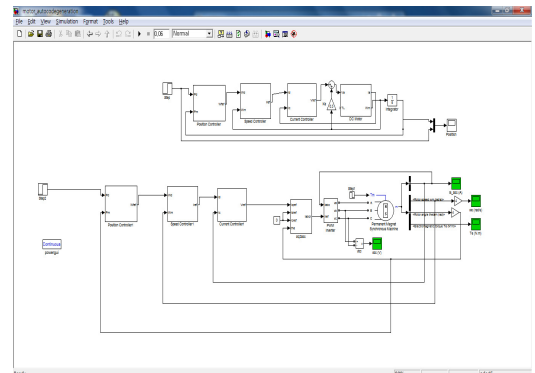


그림 7. 전체 시뮬레이션 모델  
Fig. 7. Simulation model

#### 2. 자동 코드 생성

마지막 단계는 코드 생성으로, 사용하는 마이크로컨트롤러에 탑재할 수 있도록 임베디드 타입으로 생성한다.



## V. 결론

이 논문에서는 MISRA AC SLSF 가이드를 따라서, Simulink를 사용하여, BLDC 모터 제어를 모델 기반 설계방법으로 설계하고 실제 하드웨어를 통해 동작 성능을 검증하였다. MISRA AC SLSF는 Simulink나 State Flow로 디자인된 모델기반 시스템의 안전성을 확보하기 위한 가이드로서, 이 가이드에 의거하여 디자인할 때 실제 구현 시 수행 시간의 최적화는 이루어지지 않는다. 기존의 C언어 등으로 직접 알고리즘을 구현하는 방법과 모델기반 구현을 한 후 자동 코드 생성을 통하여 구성한 제어기의 수행 시간 측정을 하였으며, 위치 제어기에서 0.8us, 속도 제어기에서 2.8us, 전류제어기에서 1.2us 등, 기존의 방법이 모든 하부 제어 루프에 있어서 수행시간은 짧은 것으로 나타났지만, 큰 차이는 보이지 않았으며 모두 제어 주기 안에 수행이 가능하였으며 마이크로컨트롤러의 성능이 꾸준히 향상됨에 따라 안정성이나 개발기간 단축 등의 장점이 있는 모델기반 제어기가 향후 제어기 구현의 주된 방법이 될 것으로 판단된다.



그림 12. 속도 제어기 수행 시간  
Fig. 12. Speed controller execution time



그림 13. 전류 제어기 수행 시간  
Fig. 13. Current controller execution time

## 참고 문헌

- [1] Asma Charfi, Chokri Mraidha, Sébastien Gérard, François Terrier, Pierre Boulet, "Does Code Generation Promote or Prevent Optimizations?", Proc. of IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing", pp.75-79, 2010
- [2] Sergey V. et. al. "Automatic Test Generation for Model-Based Code Generators", Second International Symposium on Leveraging Applications of Formal Methods, Verification and Validation, pp75-81, 2007
- [3] MISRA-C 2004, Guidelines for the use of the C language in critical systems
- [4] MISRA AC SLSF: Modeling design and style guidelines for the application of Simulink and stateflow, 2009
- [5] Seulgi Lee, Hyunwoo Chung, Hyungrok Oh,

- Junyoun Lee, "Motor control by using MATLAB/SIMULINK Auto Code Generation", KIFE conference, pp.77-78, 2012
- [6] Sanghoon Kim, "DC, AC, BLDC motor control," Bookdu, 2010
- [7] Rapid-prototype  
(<http://www.mathworks.com/rapid-prototyping>)
- [8] Jason Ghidella and Fredman, "Model-Based Design Streamlines Development of Body Electronics System", Autojournal, vol 28, no 3, pp.75-84
- [9] Jooyoung Ma, Jaemyoung Ma, Minsuk Shin, Myoungcho Sunwoo, "SILS/RCP: Integrated Model based System Development Tool for automotive embedded control system design", KASE, Spring conference, pp. 1549-1554, 2004
- [10] Ilhyung Lee, Hyochung Bang, Hyundong Oh, "PILS System Development for Collision Avoidance Algorithm Verification", KSAS Autumnal conference Vol 2, pp.1172-1175, 2008

※ 이 논문은 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (2012R1A1A2040141)

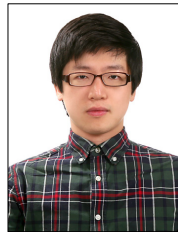
### 저자 소개

#### 신 범 식(준회원)



- 전자공학 석사
- 국민대학교 전자공학부
- <주관심분야 : 네트워크 기반 모터제어, 차량 소프트웨어, 기능안전>

#### 박 정 훈(준회원)



- 전자공학 학사
- 국민대학교 전자공학부
- <주관심분야 : 네트워크 기반 모터제어, 차량 소프트웨어, 기능안전>

#### 문 찬 우(정회원)



- 공학박사
- 국민대학교 전자공학부
- <주관심분야 : 차량전자, 지능로봇>

#### 안 현 식(정회원)



- 공학박사
- 국민대학교 전자공학부
- <주관심분야 : 차량전자, 차량 소프트웨어>