

# DSP 및 PLECS를 활용한 PMSM 구동시스템용 고속 제어 시제품개발 기법 개발

## Development of rapid control prototyping for a PMSM drive system using DSPs and PLECS

이주영\*, 최성민\*, 김세환\*\*, 이재석\*\*\*★

Jooyoung Lee\*, Sung-Min Choi\*, Sehwan Kim\*\*, Jae Suk Lee\*\*\*★

### Abstract

This paper presents implementation of rapid control prototype (RCP) for permanent magnet synchronous machines (PMSMs) using a digital signal processor (DSP) and the PLECS software. By utilization of auto code generation function in the PLECS, a current vector control algorithm for a PMSM drive system using a DSP as a control processor can be developed more efficiently. In this paper, a background of a model based design (MBD) and real time control are reviewed. Also, commercial RCP products compatible with DSP boards are introduced. At the end of the paper, experimental implementation of RCP for a PMSM drive is presented.

### 요약

본 논문에서는 Digital Signal Processor(DSP)와 PLECS 소프트웨어를 사용하여 영구자석동기전동기(Permanent Magnet Synchronous Machine, PMSM)를 위한 Rapid Control Prototyping (RCP)를 구현하였다. PLECS의 Auto-code generation (ACG) 기능을 활용하여 제어 프로세서를 DSP로 사용하는 PMSM 구동 시스템에 대한 현재 벡터 제어 알고리즘을 보다 상대적으로 간단하고 효율적으로 개발할 수 있다. 본 논문에서는 모델 기반 설계(Model Based Design, MBD)와 real time control(실시간 제어) 배경을 살펴본다. 또한, DSP 보드와 호환되는 상용 RCP 제품 및 실험을 통한 PMSM 드라이브 제어를 구현하였다.

*Key words : DSP, PMSM, Digital control, Model-based design, Rapid Control Prototype*

### 1. 서론

실시간 제어(real time control)는 많은 산업 분야에

서 제품개발 및 분석을 위해 많이 사용되고 있다 [1-3]. 실시간 제어는 컴퓨터 언어를 프로그래머가 수동으로 코딩하거나 코드 변환기에서 직접 생성되는 컴퓨터 언어

\* Graduate student, Dept. of Electrical Engineering, Jeonbuk National University

\*\* Senior researcher, Korea Institute of Machinery & Materials

\*\*\* Associate professor, Dept. of Electrical Engineering, Jeonbuk National University

★ Corresponding author

E-mail : sehwan@kimm.re.kr, jaesuk@jbnu.ac.kr, Tel : 063-270-2398

※ Acknowledgment

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.2020R1C1C1013260).

Manuscript received Mar. 31, 2022; revised Jun. 15, 2022; accepted, Jun. 27, 2022.

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

코딩 방법으로 나눌 수 있다.

디지털 신호 프로세서(Digital Signal Processor, DSP)의 사용은 실시간 제어 구현을 위한 한 가지 접근 방식이다. DSP에는 많은 기능이 통합되어 있고 다수의 아날로그 및 디지털 채널이 포함되어 있다. 따라서 많은 신호를 실시간으로 처리할 수 있다. DSP는 고성능 및 저렴한 가격과 같은 매력적인 특징이 있지만, 프로그래머 능력에 따라 컴퓨터 언어 프로그래밍 수준이 다르다. 일반적으로 DSP는 복잡성으로 인해 사용자가 DSP에 익숙해지는 데 시간이 오래 걸린다. 다른 실시간 제어 접근 방식은 Plexim RT box 또는 dSPACE microlab box 와 같은 real time machines를 사용하는 것이다. real time machine과 호환되는 소프트웨어에 컨트롤러와 시스템 모델을 포함하는 시뮬레이션 모델은 Auto-code generation(ACG) 기능을 통해 자동으로 코딩될 수 있고, 이 개발 프로세스를 Rapid Control Prototype (RCP)라고 한다. RCP를 사용하면 대상 시스템의 개발 시간을 단축하고 프로그래밍 수준의 의존성을 피할 수 있다. 그러나 상용 real time machines 비용은 DSP를 사용하는 제어 보드보다 훨씬 높으며 일반적으로 프로토타입 개발단계에서 검증 및 검증 목적으로 사용된다.

본 논문에서는 시간 및 비용 측면에서 효율적으로 구현하기 위해 PLECS 소프트웨어의 ACG 기능과 DSP를 사용하여 영구자석동기전동기(Permanent Magnet Synchronous Motor, PMSM) 드라이브용 RCP를 제안한다. 다음 섹션에서는 모델기반설계(Model Based Design, MBD) 및 RCP에 대해 간략하게 소개한다. 또한 DSP를 이용한 상용 RCP 제품도 소개된다. PMSM 드라이브용 RCP 구현 및 실험 결과도 이 논문에 나와 있다.

## II. 본론

### II-1. PMSM 구동을 위한 DSP를 사용한 Rapid Control Prototype

RCP는 그림 1의 v-chart로 표현된 MBD를 하기위한 제품개발단계 중 하나이다.

MBD 프로세스는 산업에서 효과적이고 효율적인 제품 개발을 위해 사용된다. RCP 단계에서는 시뮬레이션 된 모델을 real time machine에 연결하고 시뮬레이션 모델과 제어 알고리즘을 확인한 다음 검증하여 잠재적 오류를 최소화한다. 그러나 RCP 구현을 위한 real time machine 사용은 비용이 많이 들고 프로토타입 개발에

한계가 있다. real time machine을 DSP로 대체함으로써 보다 경제적인 상용 제품에 적용할 수 있다. 일부 제조업체는 AGC 기능이 포함된 소프트웨어와 호환되는 DSP 보드를 제공한다.

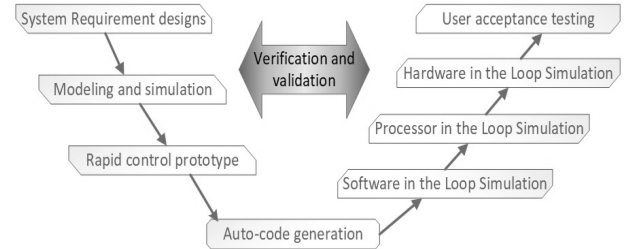


Fig. 1. MBD process in v-chart.

그림 1. v-chart에서 나타내는 MBD 프로세스

Table 1. Commercial software and TI LaunchPads for RCP implementation.

표 1. RCP 구현을 위한 상용 소프트웨어 및 TI LaunchPad

Commercial software and TI LaunchPads for RCP implementation [4-6]			
Company	PLEXIM	MathWorks	Typhoon HIL
software	PLECS	MATLAB	Typhoon HIL Control Center
DSP models	-LAUNCHXL-F28069M	-LAUNCHXL-F28027F	-LAUNCHXL-F28027F
	-LAUNCHXL-F280049C	-LAUNCHXL-F28069M	-LAUNCHXL-F28069M
	-LAUNCHXL-F28377S	-LAUNCHXL-F28377S	-LAUNCHXL-F280049C
	-LAUNCHXL-F28379D	-LAUNCHXL-F28379D	-LAUNCHXL-F28379D

Table 2. Specification of TI LaunchPads compatible for RCP implementation.

표 2. RCP 구현과 호환되는 TI LaunchPad 사양

TI Launchpad Series [7]			
Model	MCU	Pins	Feature
LAUNHXL-F28069M	TMS320F28069M	4×20	90 MHz C28x CPU, 256 KB Flash, InstaSPIN-FOC and InstaSPIN-MOTION enabled, 12-bit ADC, CAN, 2x encoder interface (eQEP), UART, SPI, and more
LAUNHXL-F28377S	TMS320F28377S	4×20	200 MHz C28x CPU, CLA, 1024 KB Flash, enabled, 12-bit or 16-bit ADCs, CAN, 24 PWM Channels, 3x encoder interface (eQEP), Sigma-delta filters, UART, SPI, and more
LAUNHXL-F28379D	TMS320F28379D	4×20	200 MHz dual C28xCPU and dual CLAs, 1 MB Flash, 16-bit or 12-bit ADCs, comparators, 12-bit DACs, delta-sigma sinc filters, HRPWMS, eCAPs, eQEPs, CANs and more

표 1은 TI(Texas Instrument) LaunchPad DSP 보드 및 보드와 호환되는 소프트웨어를 보여준다.

표 2는 RCP 구현을 가능하게 하는 LaunchPad 모델들의 특징을 요약한 것이다.

본 논문에서는 PMSM이 고효율 및 전력 밀도 특성으로 인해 운송 및 가전제품과 같은 많은 산업 응용 분야에 활용되어 왔기 때문에 RCP를 적용하여 PMSM 제어 시스템을 구현하였다.

II-2. DSP와 PLECS를 이용한 RCP 구현

TI LaunchPad는 마이크로컨트롤러 초급 사용자를 위해 설계된 저비용 및 완전한 기능을 갖춘 개발 플랫폼이다[8]. LaunchPad는 대상 애플리케이션을 개발하는 동안 표준화되고 사용에 편리한 플랫폼을 제공한다. 또한, LaunchPad는 산업 및 자동차 애플리케이션에서 디지털 전력 및 모터 제어를 가능하게 하는 실시간 MCU이다.

Table 3. LAUNCHXL-F28379D.

표 3. LAUNCHXL-F28379D

LAUNCHXL-F28379D [8]	
MCU	DSP : TMS320F28379D
JTAG	USB connected isolated XDS100v2
Pins	4×20 pins
Connectors	- Two encoder interface - Isolated CAN transceiver
Number of BoosterPacks available	2 Plug-in Modules

표 3은 PLECS 소프트웨어와 호환되는 LaunchPad 모델 중 하나인 LAUNCHXL-F28379D의 기능을 요약한 것으로 본 실험에서 사용된 LaunchPad이다. RCP 구현의 경우 LaunchPad는 real time machines으로 작동한다.

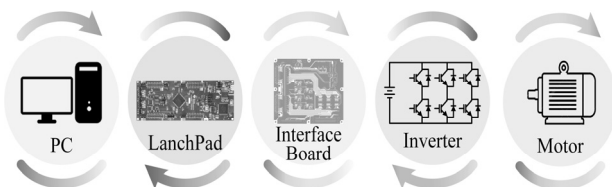


Fig. 2. The microcontroller system flow.

그림 2. 마이크로컨트롤러 시스템 흐름도

그림 2는 RCP 구현을 위해 설계된 마이크로컨트롤러 시스템의 전체 흐름을 보여준다. 사용자는 컴퓨터로 시

스템을 설계하고, 설계된 시스템은 컨트롤러, 인터페이스 보드 및 인버터의 신호를 주고받아 타겟을 제어한다.

그림 2의 인터페이스 보드는 인버터와 컨트롤러의 신호를 연결하는 역할을 한다. 모터 제어를 하기 위해서 인버터가 컨트롤러로부터 Gate 신호를 전달받아야 하고 컨트롤러는 인버터로부터 피드백 신호를 받아야 한다. 이때 제어기의 디지털 입력, 디지털 출력, 아날로그 입력의 전압 범위는 고정되어 있으므로 인버터와 제어기 사이의 신호를 전압 레벨에 맞게 연결하는 인터페이스 보드를 설계해야 한다. 이뿐만 아니라 모터의 엔코더 신호, 토크 센서 신호 및 각종 외부 신호는 인터페이스 보드를 통해 컨트롤러로 전달될 수 있다.

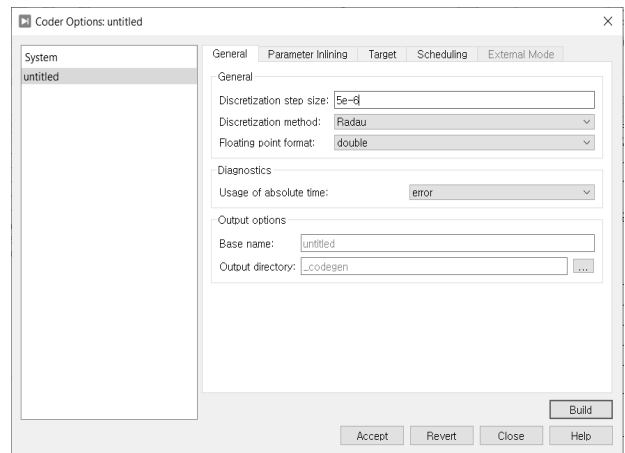


Fig. 3. PLECS Coder option window for setting.

그림 3. PLECS Coder option 설정 창

그림 3은 LaunchPad와 PLECS 소프트웨어를 사용하여 RCP를 구현할 때 시스템을 설정하는 창을 나타낸다. 이 창을 통해 사용자는 시스템 step-size, LaunchPad의 종류, Build 유형, 외부 모드 등 설계된 시스템이 LaunchPad로 빌드 될 때 필요한 정보들을 설정할 수 있다.

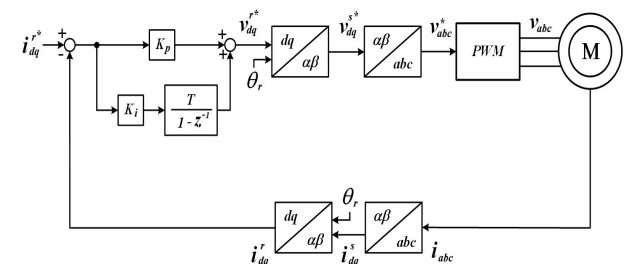


Fig. 4. The microcontroller system flow.

그림 4. PMSM 제어시스템 흐름도

그림 4는 비례-적분(Proportional-integral, PI) 제어기를 사용하여 개발된 전류 벡터 제어기를 포함하는 PMSM 제어시스템의 블록도이다. 그림 5는 PLECS 소프트웨어를 사용하여 설계된 PMSM 구동의 현재 벡터 제어시스템을 보여준다.

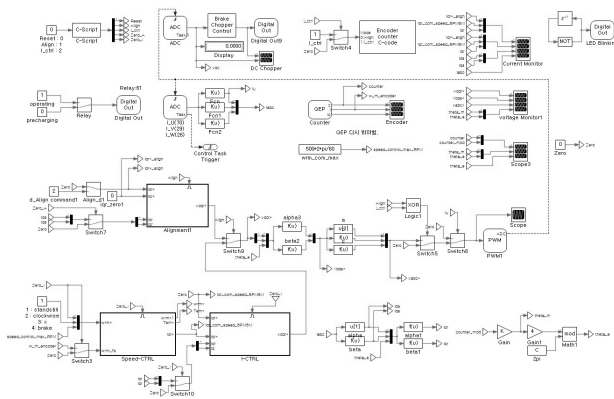


Fig. 5. Designed system in PLECS.  
그림 5. PLECS로 설계된 시스템

```

/*
 * Header file for: speed_current_ctrl_for_SPMSM_dsp_V1
 * Generated with : PLECS 4.4.6
 * TI2837xS 1.1
 * Generated on : 18 Aug 2021 20:58:54
 */
#ifndef PLECS_HEADER_speed_current_ctrl_for_SPMSM_dsp_V1_h_
#define PLECS_HEADER_speed_current_ctrl_for_SPMSM_dsp_V1_h_

#include <stdbool.h>
#include <stdint.h>

/* Target declarations */
typedef int_fast8_t int8_t;
typedef uint_fast8_t uint8_t;
extern void speed_current_ctrl_for_SPMSM_dsp_V1_background(void);

/* Model floating point type */
typedef float speed_current_ctrl_for_SPMSM_dsp_V1_FloatType;

/* Model checksum */
extern const char * const speed_current_ctrl_for_SPMSM_dsp_V1_checksum;

/* Model error status */
extern const char * speed_current_ctrl_for_SPMSM_dsp_V1_errorStatus;
    
```

Fig. 6. Generated C-code through ACG.  
그림 6. ACG로 생성된 C-code

그림 6은 그림 5의 시스템을 ACG 기능을 통해 자동으로 생성된 C-code를 보여준다. PLECS에서 시스템을 설계하고, Coder 옵션 창에서 설정하고 빌드하면 자동으로 생성된 C-code가 LaunchPad에 다운로드 된다.

II-3. 실험 결과

그림 7은 TI LaunchPad DSP와 PLECS 소프트웨어를 사용하여 PMSM 드라이브를 제어하기 위한 RCP 구현을 위한 실험셋업을 보여준다. 실시간 시뮬레이션 및 ACG가 가능한 PLECS 소프트웨어와 LAUNCHXL-

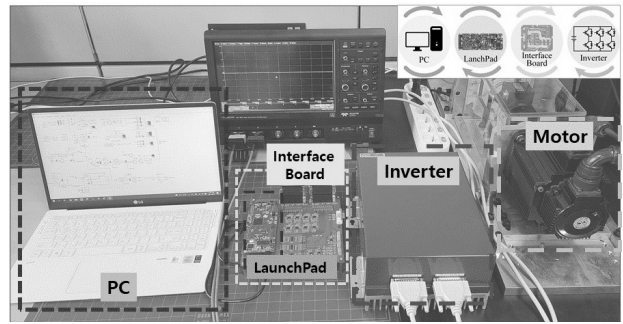


Fig. 7. The set-up for test implementation.  
그림 7. 테스트 실험 셋업

F28379D와 PMSM을 사용하였다.

표 4와 5는 실험에 사용된 모터와 인버터의 사양을 요약한 것이다.

Table 4. The motor specifications.

표 4. 모터 사양

Motor Specification	
Rated Output Power	1.5 [kW]
Power Voltage	200 [VAC]
Rated Torque	7.16 [Nm]
Rated Current	9 [Arms]
Rated Speed	2000 [rpm]
Rotor Moment of Inertia*	$14.5 \times 10^{-4} [kg \cdot m^2]$

Table 5. The inverter specifications.

표 5. 인버터 사양

Inverter Specification	
DC link Maximum Voltage	750 [V]
Switching Frequency	10 [kHz]
Output Current	30 [A]
Driver type	IGBT

DSP 및 LaunchPad를 사용하여 RCP 구현을 위한 초기 실험 테스트를 위해 그림 8과 같이 디지털 및 아날로그 I/O 채널 신호점검 실험을 진행하였다. 실험에 사용될 아날로그 채널에 신호를 연결하고 PLECS로 확인하였다. ±7.5V AC 신호는 인터페이스 보드를 거쳐 0.3~2.8V로 바뀌었고 아날로그 채널에 연결되었다. 채널을 걸쳐 나온 그림 8의 신호는 0.28~2.73V로 2.5~2.9% 정도의 오차가 발생하였다. 오차의 발생 원인은 외부 환경 요인 등 다양한 원인이 있으며 Function generator, Oscilloscope, LaunchPad에서 발생하는 오차율 또한 감안해야 한다.

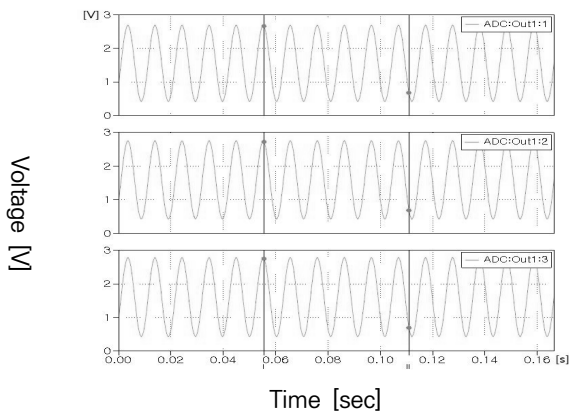


Fig. 8. Experimental verification of ADC signal.  
 그림 8. ADC 신호 검증 실험결과

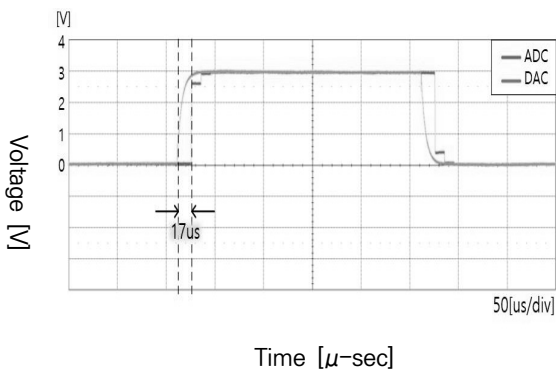


Fig. 9. Experimental verification of ADC-DAC time delay.  
 그림 9. ADC-DAC 시간 지연 실험결과

그림 9는 PLECS와 LaunchPad를 사용하여 RCP를 구현할 때 나타나는 ADC-DAC 시간 지연을 보여준다. 일정한 주기의 펄스 신호가 ADC로 입력되고, 이 신호를 DAC(Digital to Analog Conversion)로 출력하였다. 그림 9와 같이 약  $17[\mu\text{-sec}]$ 의 지연이 발생함을 실험을 통해 확인하였다. ADC-DAC의 시간 지연은 실시간 시뮬레이션에서 한 주기 동안 신호와 제어에 사용되는 신호의 시간 차이를 나타낸다. 즉, 실험 시 사용할 데이터 샘플링 주기를 결정하는데 그림 9의 시간지연 실험결과를 활용가능하다. 본 실험에서 한 주기 시간은 PWM carrier frequency 등을 고려해  $100[\mu\text{-sec}]$ 를 사용하였다.

그림 10과 11의 실험 결과는 PLECS와 DSP를 사용하여 RCP를 통한 PMSM 드라이브의 전류 벡터 제어를 보여준다. 그림 4에 보이는 바와 같이 PI 제어기를 활용하여 전류벡터제어시스템을 구현하였다. 본 실험에서는 제안하는 기법을 이용한 d축과 q축 전류의 독립적인 제어 가능여부에 대한 검증을 진행하였다. 그림 10은 모터가

초기 회전자 위치에 정렬되었을 때 d축 전류제어의 실험 파형을 보여준다.

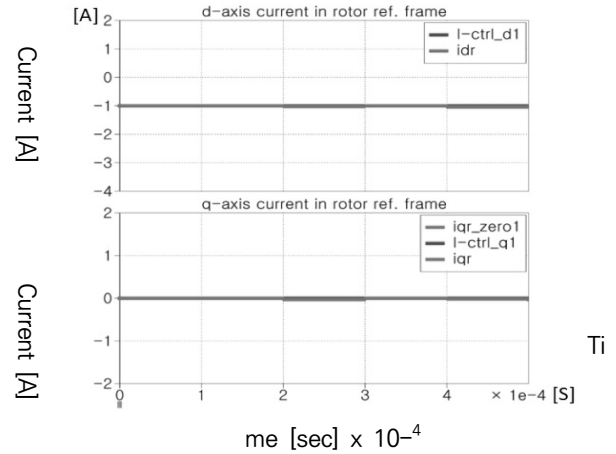


Fig. 10. Experimental results of the d-axis current control of the PMSM.

그림 10. PMSM d축 전류제어 실험 결과

그림 10에 보이는 바와 같이 d축에 전류를 인가하여 PMSM의 회전자를 초기위치에 정렬시키고 q축 전류는 0[A]로 각각 제어됨을 실험을 통해 확인하였다.

다음으로는 q축 전류제어실험을 진행하였다. PMSM의 회전자를 초기위치에 정렬시킨 후 d축에는 0[A] q축에는 1[A]의 지령을 각각 인가하였다. 그림 11에 보이는 바와 같이 제안하는 방법을 통하여 모터의 d축 전류와 q축 전류가 독립적으로 제어가 됨을 실험을 통해 검증하였다.

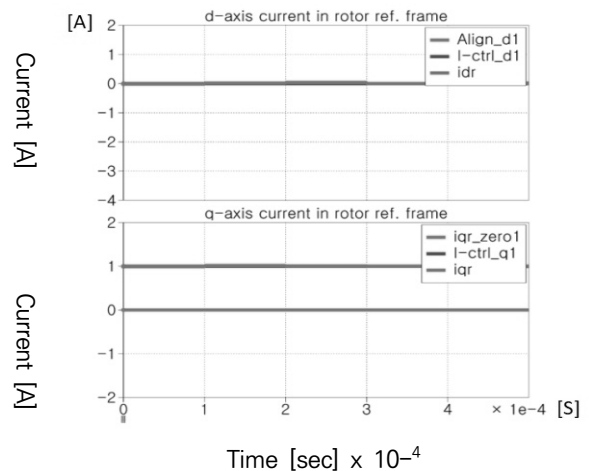


Fig. 11. Experimental results of the q-axis current control of the PMSM.

그림 11. PMSM q축 전류제어 실험 결과

### III. 결론

본 논문에서는 PMSM 드라이브용 RCP의 구현과 실험 결과를 보여준다. DSP 및 ACG기능을 활용한 RCP 구현을 통해 PMSM 제어시스템을 시간 및 비용 측면에서 효율적인 개발이 가능하다. 또한, real time machine을 대체하여 DSP가 적용된 LaunchPad를 사용함으로써 비용면에서 효율적인 PMSM 제어시스템 개발 및 구현하였고 PMSM의 전류제어실험을 통하여 제안한 PMSM 드라이브용 RCP를 검증하였다.

### References

- [1] I. R. Kendall and R. P. Jones, "An investigation into the use of hardware-in-the-loop simulation testing for automotive electronic control systems," *Control Eng. Pract.*, vol.7, no.11, pp.1343-1356, 1999. DOI: 10.1016/S0967-0661(99)00103-3
- [2] T. N. Chang, B. Cheng, and P. Sriwilaijaroen, "Motion control firmware for high-speed robotic systems," *IEEE Trans. Ind. Electron.*, vol.53, no.5, pp.1713-1722, 2006. DOI: 10.1109/ICMECH.2004.1364415
- [3] L. Gauchia and J. Sanz, "A per-unit hardware-in-the-loop simulation of a fuel cell/battery hybrid energy system," *IEEE Trans. Ind. Electron.*, vol.57, no.4, pp.1186-1194, 2010. DOI: 10.1109/TIE.2009.2036641
- [4] Plexim, "RT box Launchpad Interface," [https://www.plexim.com/products/rt\\_box/launchpad\\_interface](https://www.plexim.com/products/rt_box/launchpad_interface).
- [5] Mathworks, "Videos and Webinars-Getting Started with Embedded Coder Support Package for TI C2000 Processors," <https://kr.mathworks.com/videos/getting-started-with-embedded-coder-support-package-for-ti-c2000-processors-1573540550102.html>.
- [6] Typhoon HI, "LaunchPad Interface board" <https://www.typhoon-hil.com/products/launchpad-interface-board>.
- [7] C2000 real-time control MCUs | Design & development | TI.com, "C2000 real-time

Microcontrollers - design & development" <https://www.ti.com/microcontrollers-mcus-processors/microcontrollers/c2000-real-time-control-mcus/design-development.html/>

[8] Texas Instruments Incorporated, "LAUNCHXL-F28379D Overview," Texas Instruments Corp., Dallas, Texas, USA, Tech. Rep. SPRUI77C, 2019.

### BIOGRAPHY

#### Jooyoung Lee (Member)



2020 : BS in Electrical Engineering, Jeonbuk National University.  
2020~Present : MS candidate student at the Department of Electrical Engineering, Jeonbuk National University.

#### Sung-Min Choi (Member)



2018 : BS in Electrical Engineering, Kyungnam University.  
2020 : MS in Electrical Engineering, Jeonbuk National University.  
2020~Present : Ph.D. candidate student at the Department of Electrical Engineering, Jeonbuk National University.

#### Sehwan Kim (Member)



2011 : BS in Electrical Engineering, Yeungnam University.  
2016 : PhD in Electrical Engineering, Yeungnam University.  
2016~2018 : Research Engineer, POSCO International Co., Ltd.  
2018~2019 : Research engineer, Doosan Infracore Co., Ltd.  
2019~Present : Senior researcher, Korea Institute of Machinery & Materials.

**Jae-Suk Lee** (Member)

2006 : BS in Electrical Engineering,  
Inha University.

2009 : MS in Electrical Engineering,  
the University of Wisconsin-Madison.

2013 : PhD in Electrical Engineering,  
the University of Wisconsin-Madison.

2014~2015 : Electrical engineer,  
General Electric global research,  
USA.

2015~2017 : Assistant Professor in Department of  
Electrical Engineering, Kyungnam University.

2017~Present : Associate Professor at the Department  
of Electrical Engineering, Jeonbuk National University.