

The World Leader in High Performance Signal Processing Solutions



# Cortex-M4F를 기반으로 한 ADSP\_CM40x와 Simulink를 이용한 교류 전동기 제어

2014년 10월 14일 임종수(fermi.lim@analog.com)

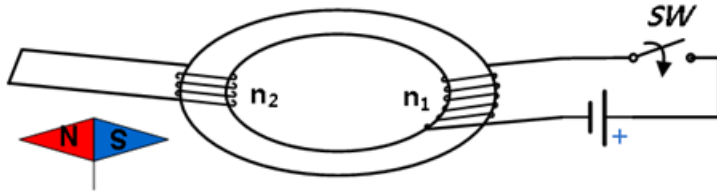
# Topics

1. 교류 전동기 동작 원리 및 제어 방법.
2. 교류 전동기 구동 시스템.
3. 교류 전동기 구동을 위한 MCU 사양.
4. ADSP\_CM40x 사양 : QEP, PWM, ADC
5. IAR, KEIL 기반의 device drivers.
6. Simulink를 이용한 vector 제어 model.
7. Simulink로 생성한 Code로 motor 제어.
8. 기 타.

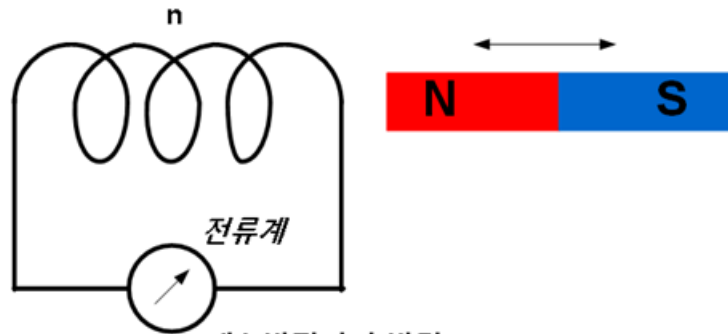
# 교류 전동기 동작 원리 (1).

◆ 얘기의 시작 ...

➤ 1831 Faraday 2 :



(a) 교류의 발견



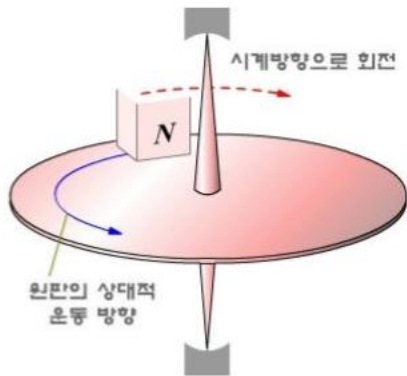
(b) 발전기의 발명

[그림 1.2-1] Faraday의 실험.

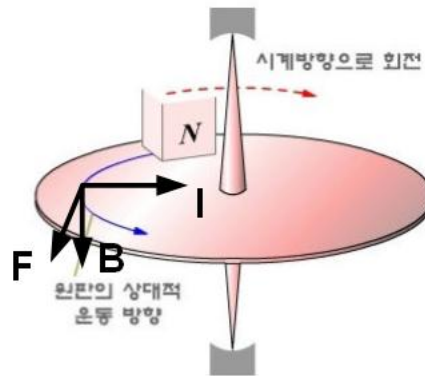
➤ (EMF)

# 교류 전동기 동작 원리 (2).

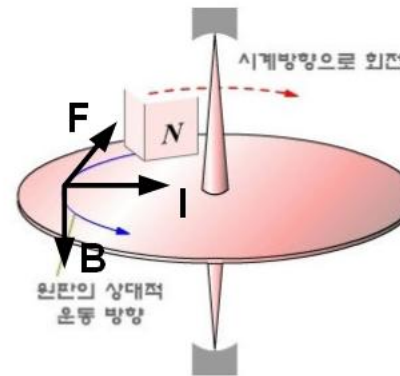
## ◆ Arago의 원판 :



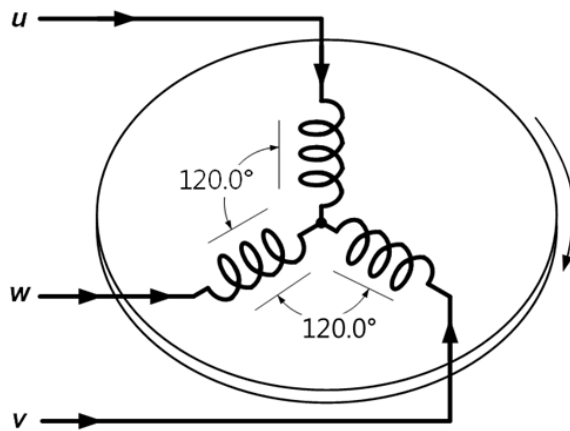
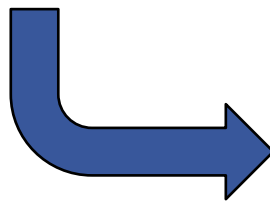
Arago의 원판



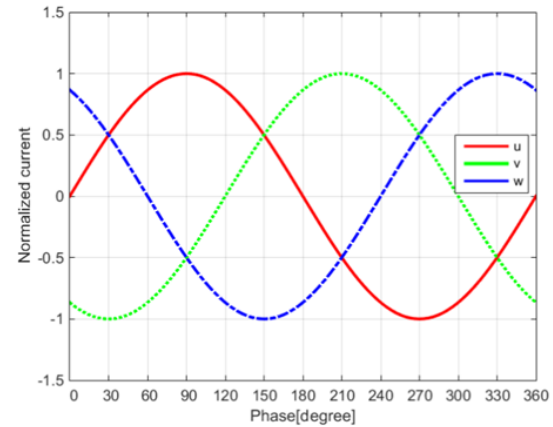
플레밍의 오른손 법칙



플레밍의 왼손 법칙



(a) u, v, w Y 결선



(b) u, v, w 3상 전류

# 교류 전동기 동작 원리 (3).

## ◆ 유도 전동기 : Induction Motor

- slip torque
- model : 3, 3.
- 6 .

## ◆ 동기식 전동기 : Permanent Magnetic Syn. Motor (Permanent Magnet)

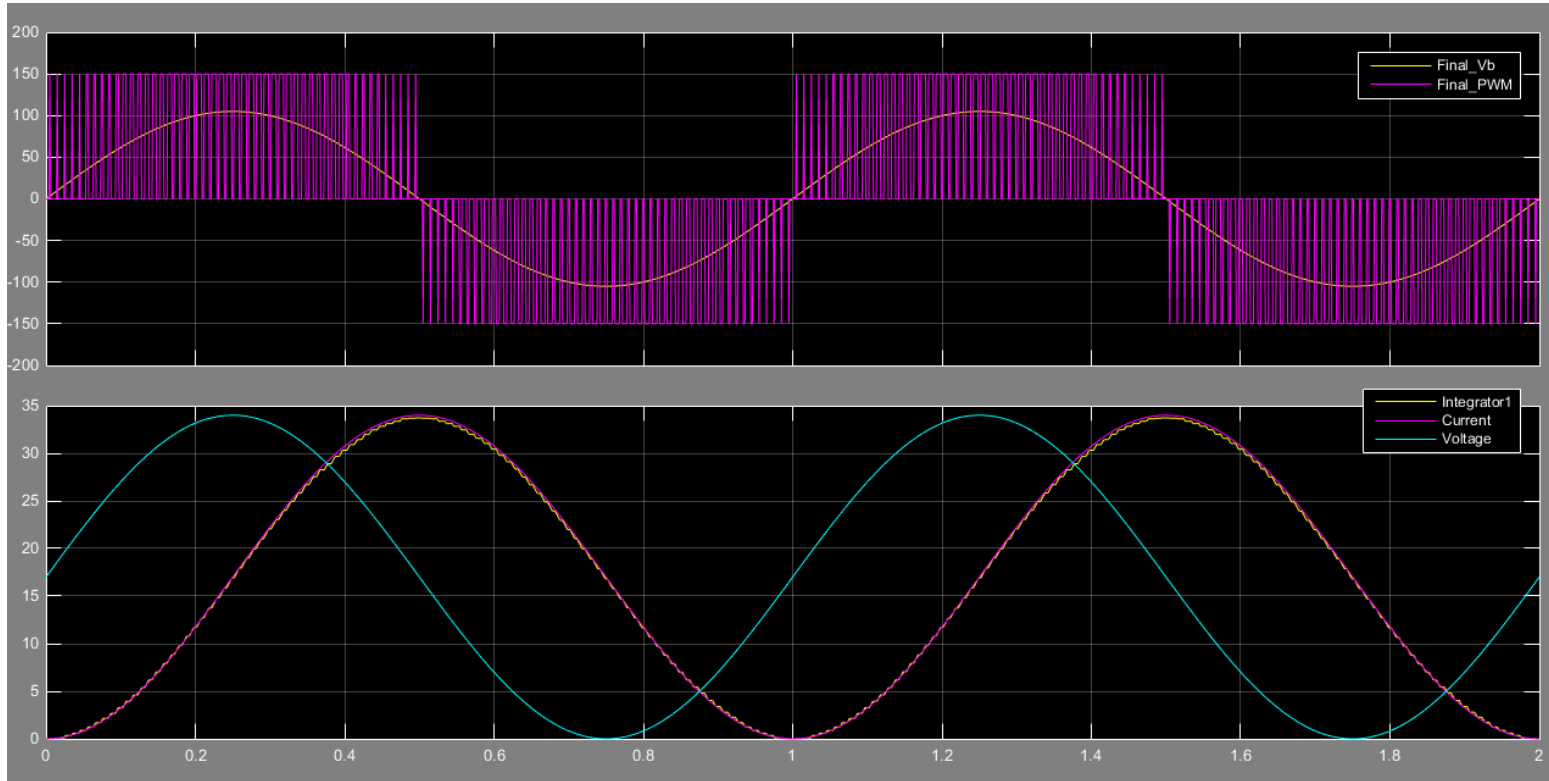
- .
- .
- model : 3, 3 .

## ◆ 대용량 전력이 필요 :

- IGBT, MOSFET, switching
- PWM port .

# 교류 전동기 동작 원리 (4).

## ◆ 전력용 반도체를 위한 PWM :



# 교류 전동기 제어 방법(1).

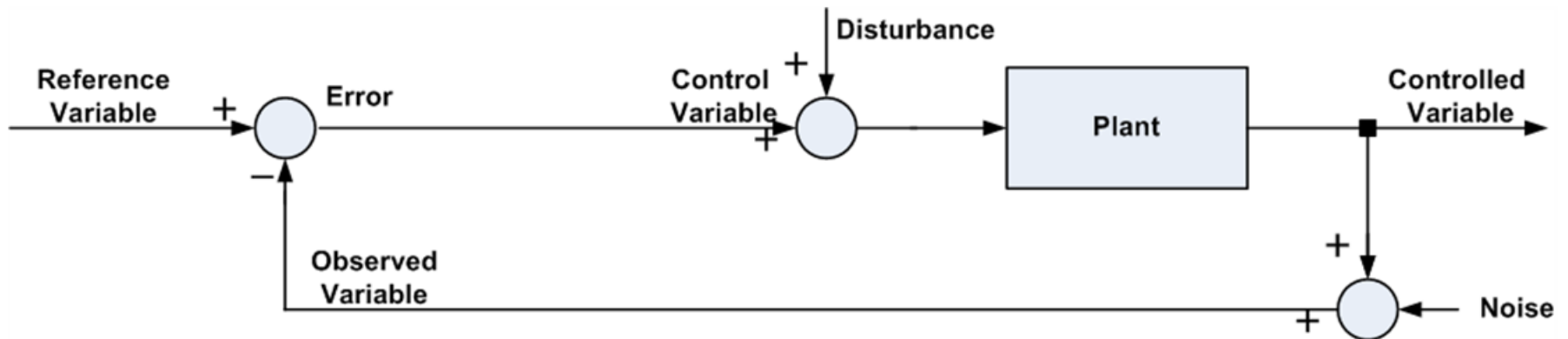
## ◆ System :

### ➤ Open System :



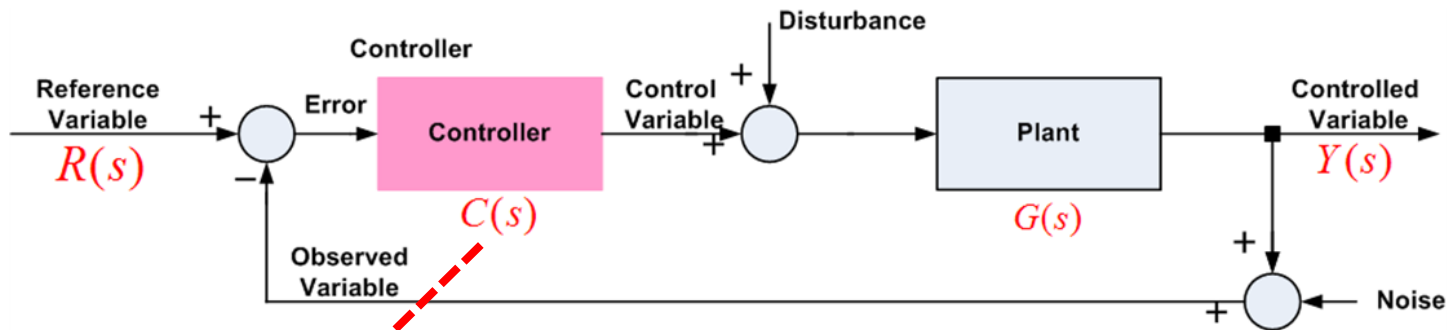
[그림 2.3.3-1] open system.

### ➤ Closed System :

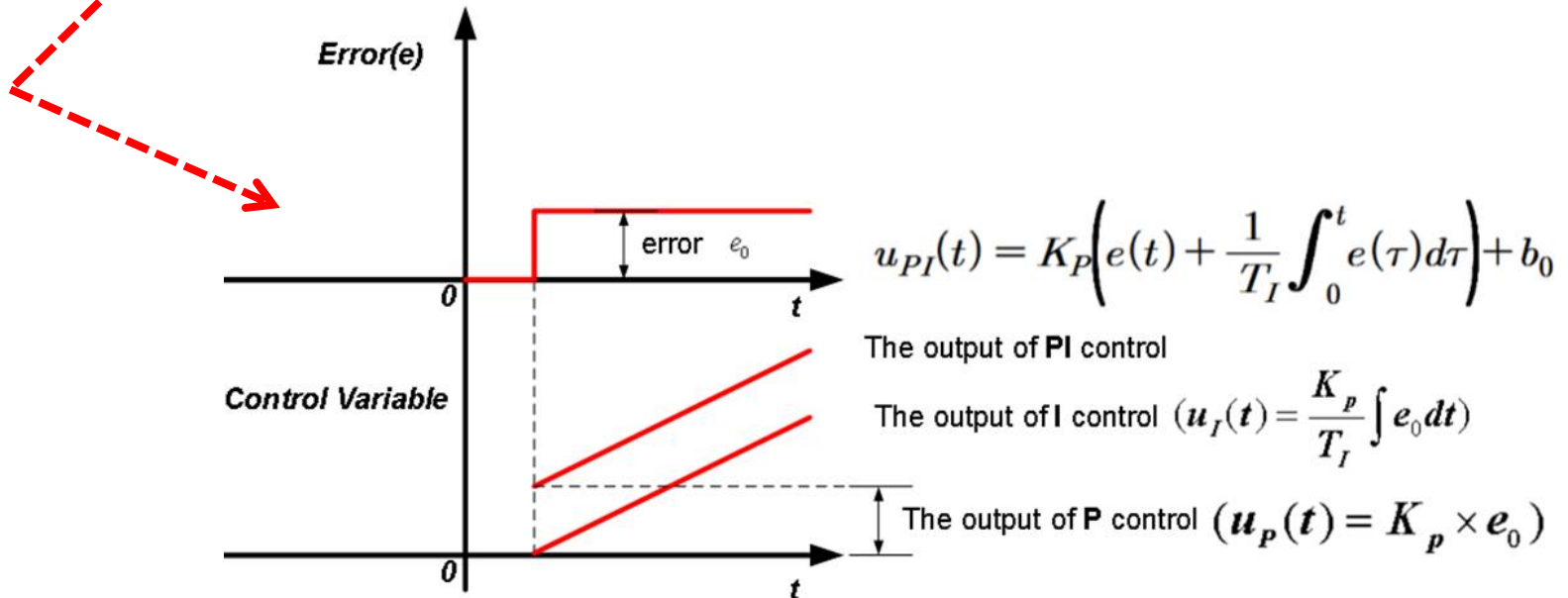


# 교류 전동기 제어 방법(2).

## ◆ Closed System with controller :

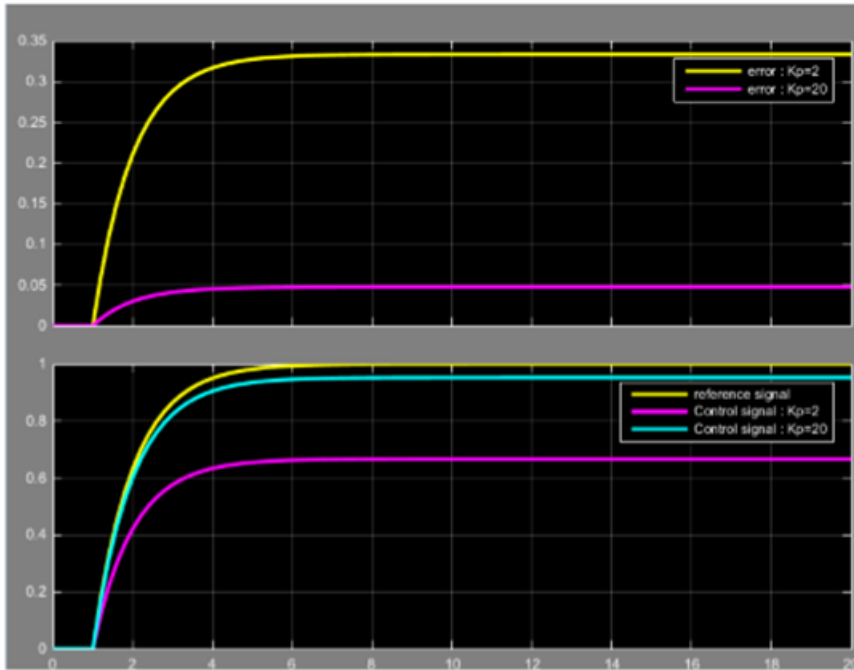


[그림 2.3.3-3] closed-loop system.

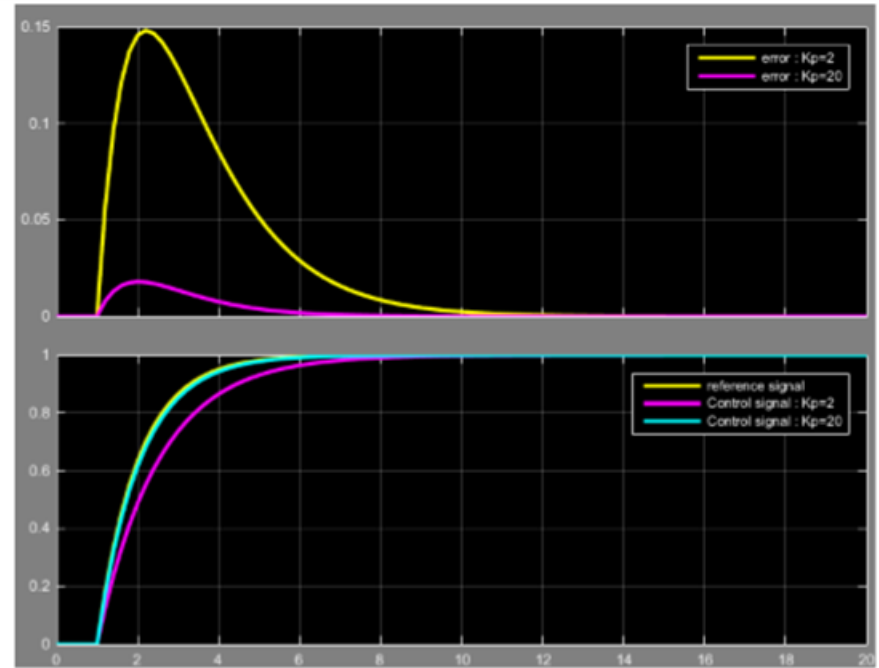


# 교류 전동기 제어 방법(3).

## ◆ Closed System with PI controller :



(a) P 제어기 출력.

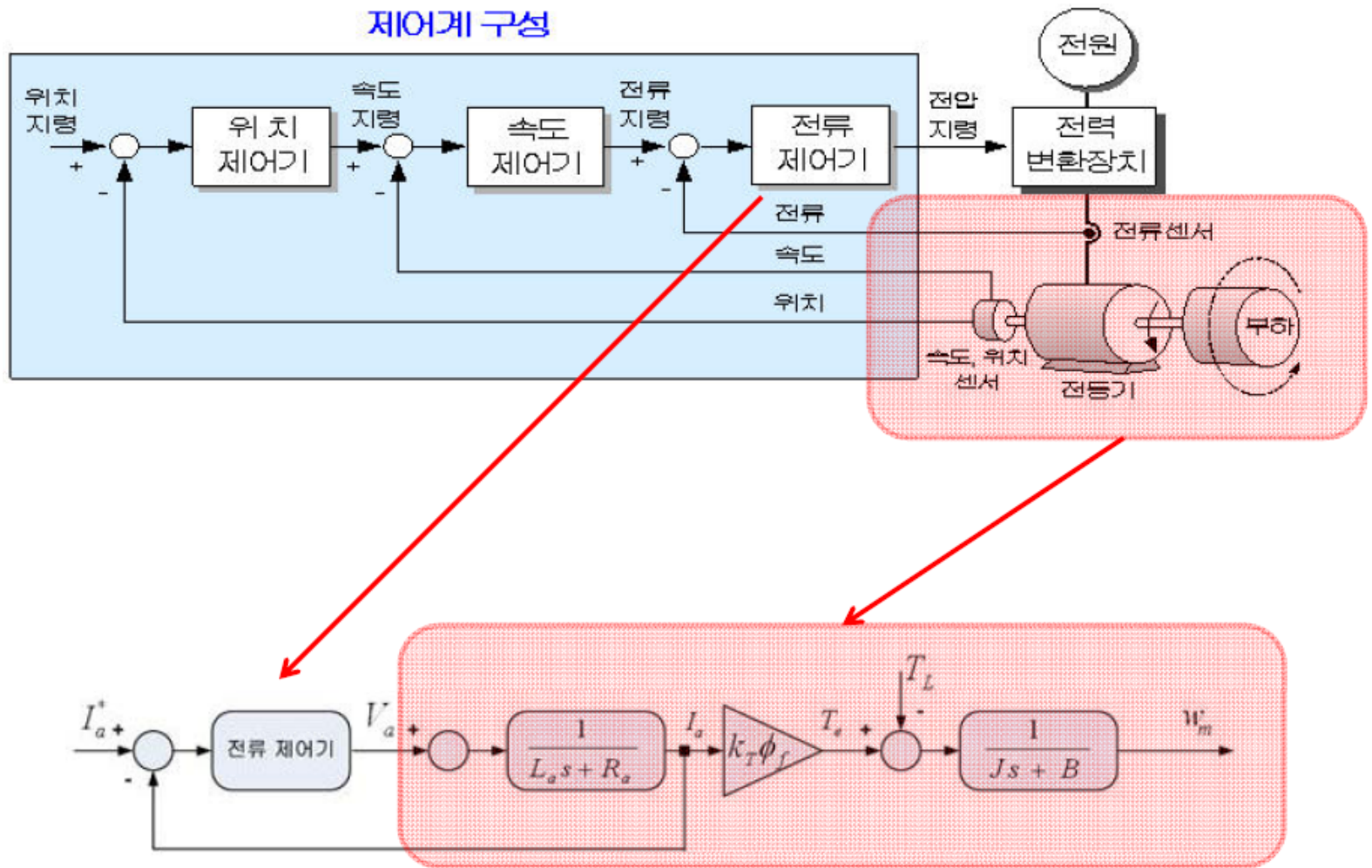


(b) PI 제어기 출력.

[그림 2.3.3-10] P제어기와 PI 제어기 비교.

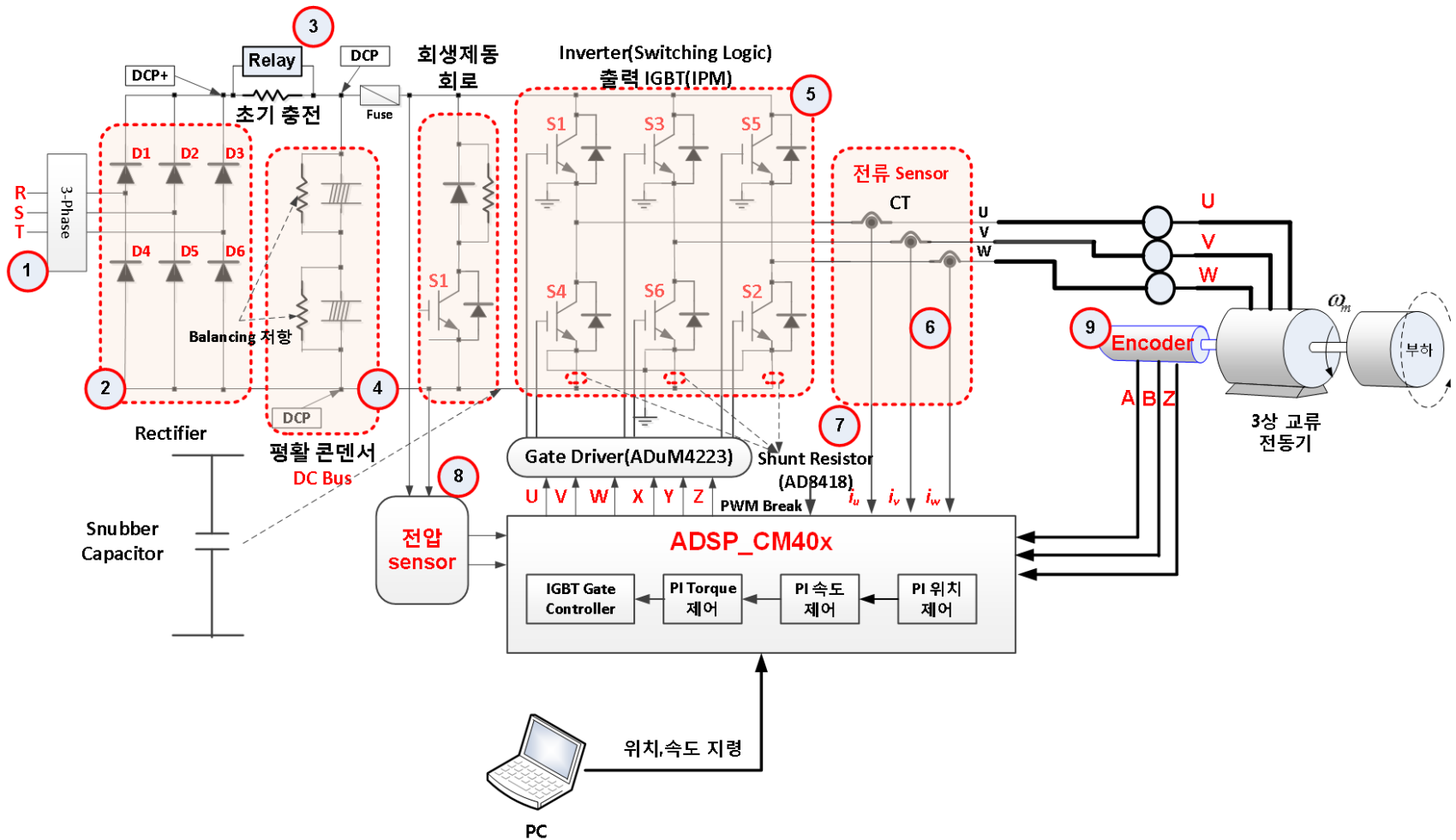
# 교류 전동기 제어 방법(4).

- ◆ 교류 전동기 제어를 위한 구성도 :



# 교류 전동기 구동 시스템.

## ◆ 3상 교류 전동기 구동 system의 구조 :



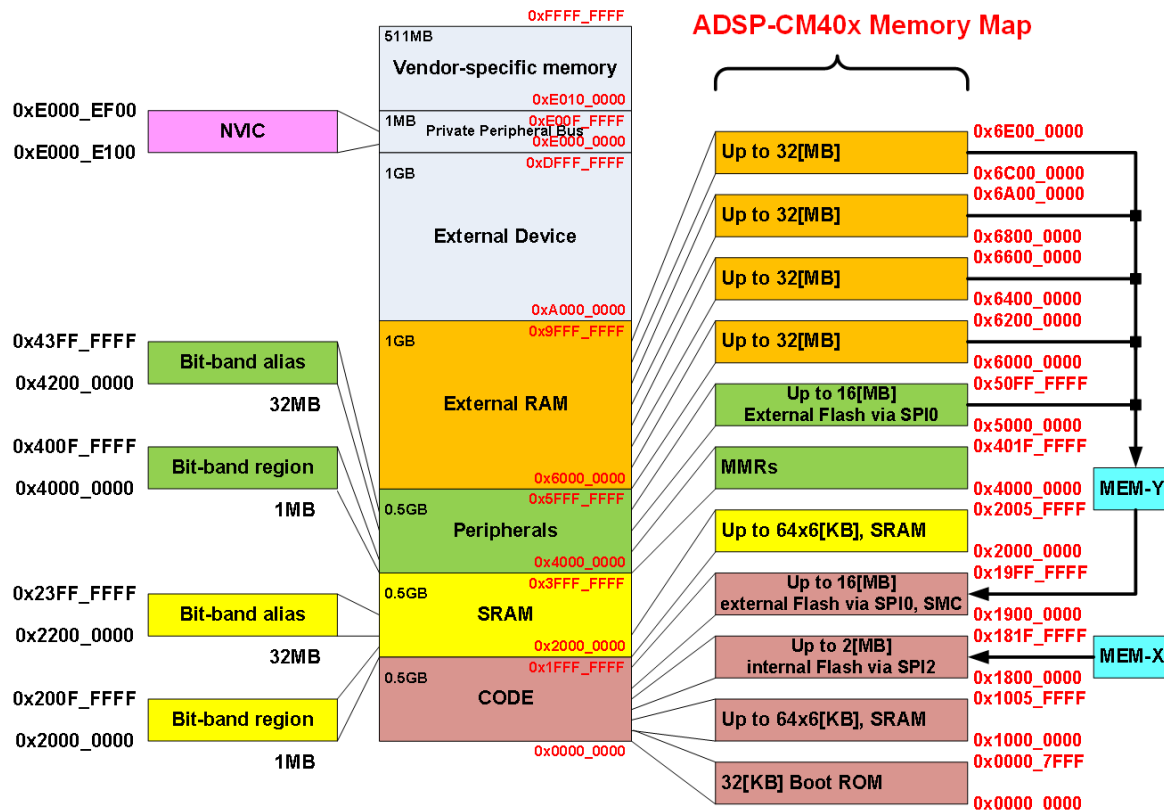
# 교류 전동기 구동을 위한 MCU 필요 사항.

- ◆ 회전자의 위치 및 속도 측정 :
  - Resolver : analog digital values . → **AD2S12xx RDC**
  - Encoder : **Incremental encoder** Absolute encoder.
- ◆ 전류 및 전압 측정 :
  - ADC : 2 3 , 1 DC link ADC ports.
- ◆ PWM Switching : 교류 전동기 제어를 위한 **전용** PWM ports.
  - PWM : PWM duty **dead-time** .
- ◆ 기타 :
  - :
    - 2 , 2 , 3
    - FPU .
  - DSP Software Library :
    - filters DSP library → **CMSIS**
  - profiler interfaces :
    - profiler interfaces GPIOs UART, CAN .
  - **Trip zone** port .

# ADSP\_CM40x 사양 : Cortex-M4F.

## ◆ Cortex-M4 :

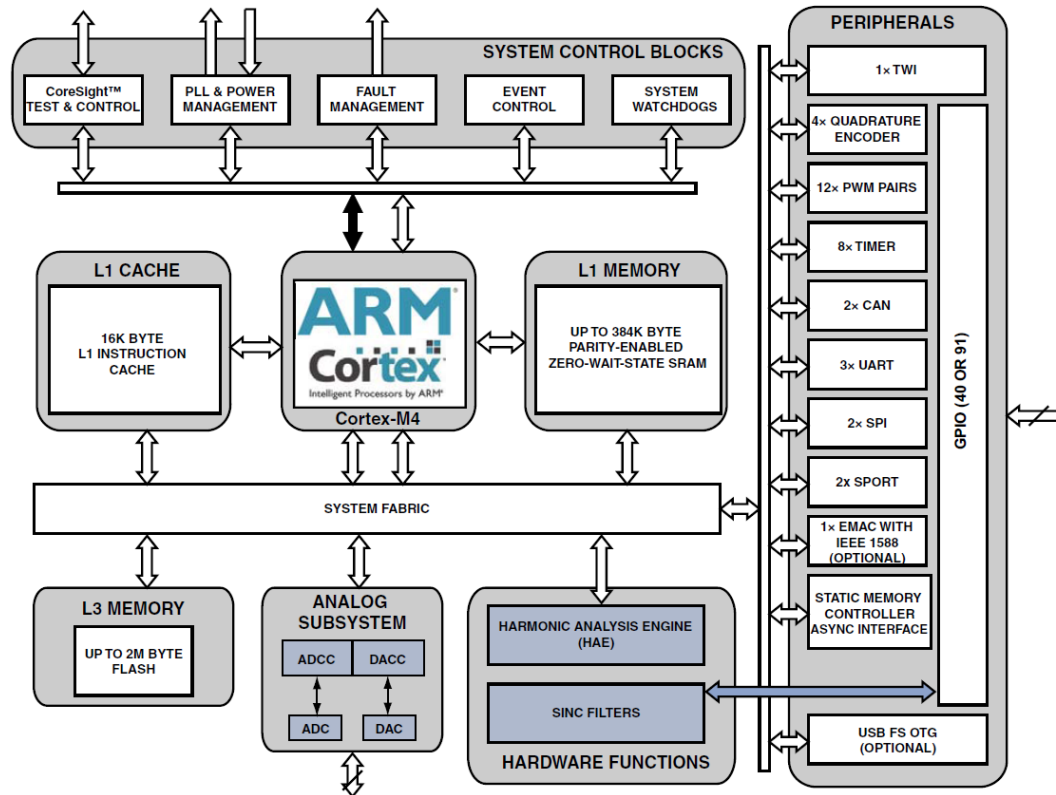
- interrupt controller : NVIC
- Thumb-2 instruction set : ARM mode
- memory map : FPU



# ADSP\_CM40x 사양(1)

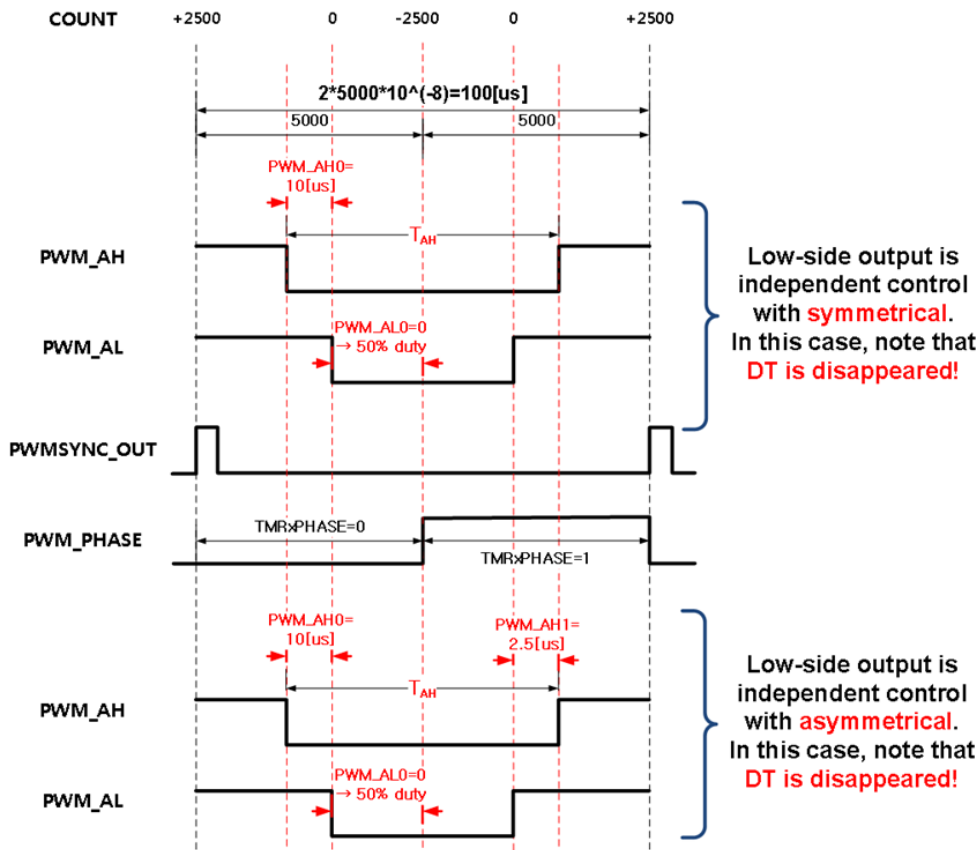
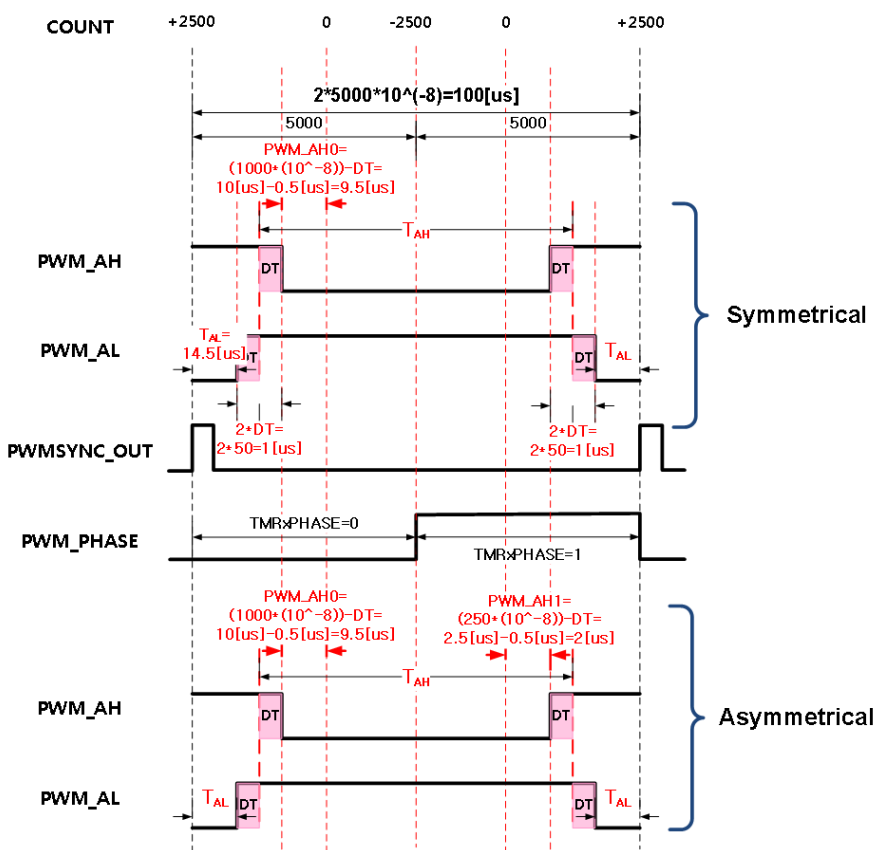
## ◆ ADSP\_CM40x :

- 384[KB] SRAM, 2[MB] internal Flash, 240[MHz] Core clock
- 4 QEPs : system clock encoder pulses counting.
- system clock = 50~100[MHz]



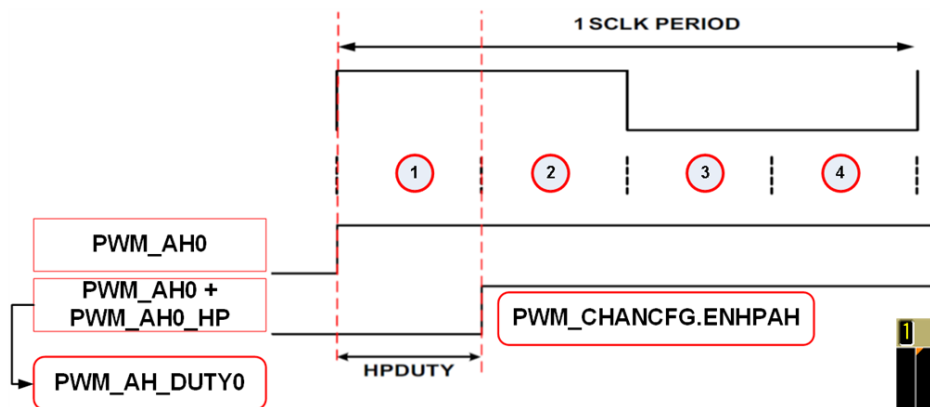


# ADSP\_CM40x 사양(2) : PWM(2)

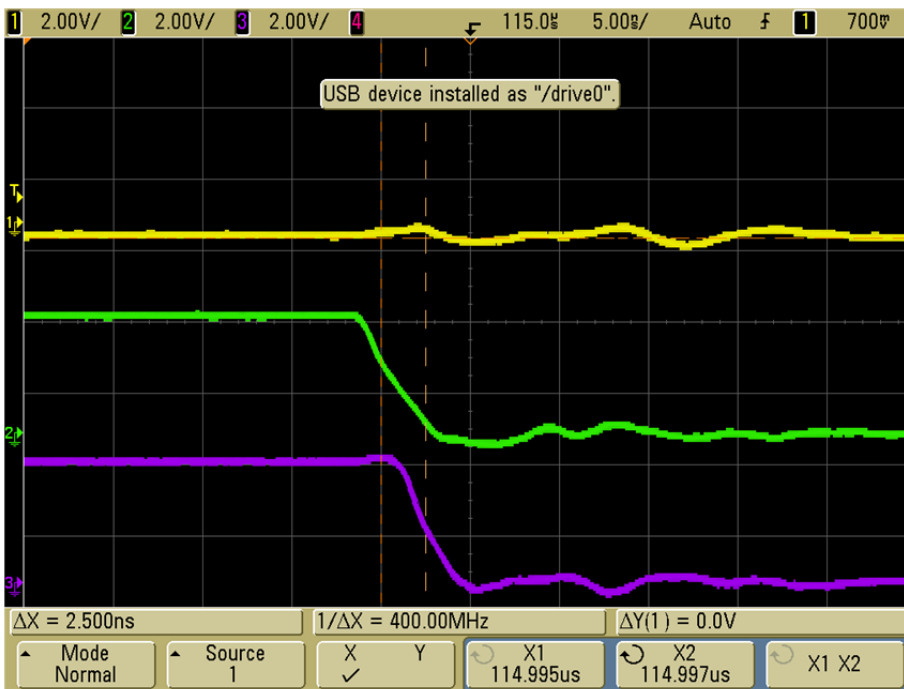
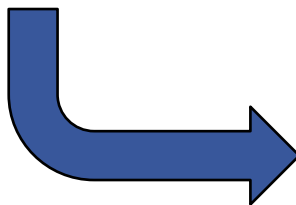


# ADSP\_CM40x 사양(2) : PWM(3)

- ◆ HPPWM(Heightened Precision PWM) : system clock **1/4**까지 duty 제어



[그림 19.2-8]  $T_{sysclk}$ 를 4등분하여 duty cycle 제어.



# ADSP\_CM40x 사양(2) : ADC(1)

## ◆ ADSP\_CM40x : ADC ports

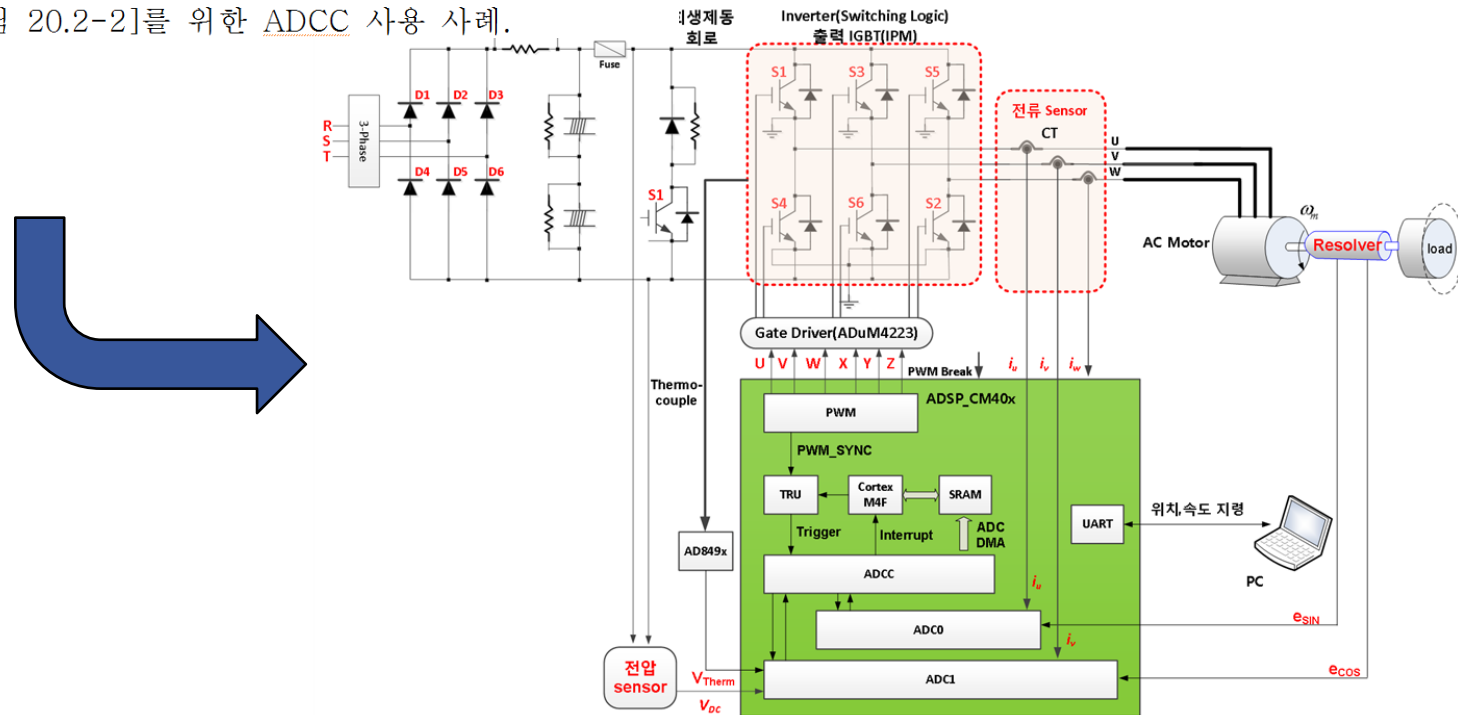
- 2 16-bit SAR A/D Converter
  - CM402F/CM403F : 24 channels, CM407F/CM408F : 16 channels
  - 2.66[MSPS], 375[ns] conversion rate.
  - event base AD conversion :
    - event time, event control, event data, event status.
    - 2 SAR ADC **sampling**
  - 2 event interrupts 1 ADC error status interrupt
  - TRU : trigger master/slaver
    - **6** Trigger slave
    - **2** Trigger master



# ADSP\_CM40x 사양(2) : ADC(3)

event	Timer	ADC	ADC Channel	Event Timer	요구 사항
$I_u$	Timer 0	ADC 0	ADC0_00	0	동시 sampling
$I_v$	Timer 1	ADC 1	ADC1_03	0	
$V_{DC}$	Timer 0	ADC 1	ADC1_01	$t_{E2}$	
$V_{Therm}$	Timer 1	ADC 1	ADC1_00	$t_{E3}$	
$e_{SIN}$	Timer 1	ADC 0	ADC0_01	$t_{E0}$	동시 sampling
$e_{COS}$	Timer 1	ADC 1	ADC1_02	$t_{E1} = t_{E0}$	

[표 20.2-1] [그림 20.2-2]를 위한 ADCC 사용 사례.







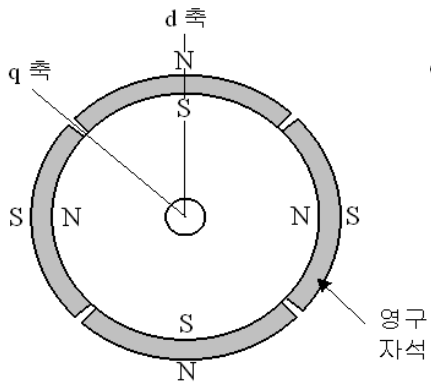
# Simulink를 이용한 vector 제어 model(1)

## ◆ SPMSM 분석 :



Wr

d-q



$$\frac{d}{dt} i_{ds}^w = \frac{1}{L_s} (v_{ds}^w - R_s i_{ds}^w + \underline{w_r L_s i_{qs}^w})$$

$$\frac{d}{dt} i_{qs}^w = \frac{1}{L_s} (v_{qs}^w - R_s i_{qs}^w - \underline{w_r L_s i_{ds}^w} - w_r \phi_f)$$



torque

$$T = \frac{P}{2} \frac{3}{2} \phi_f i_{qs}^r$$



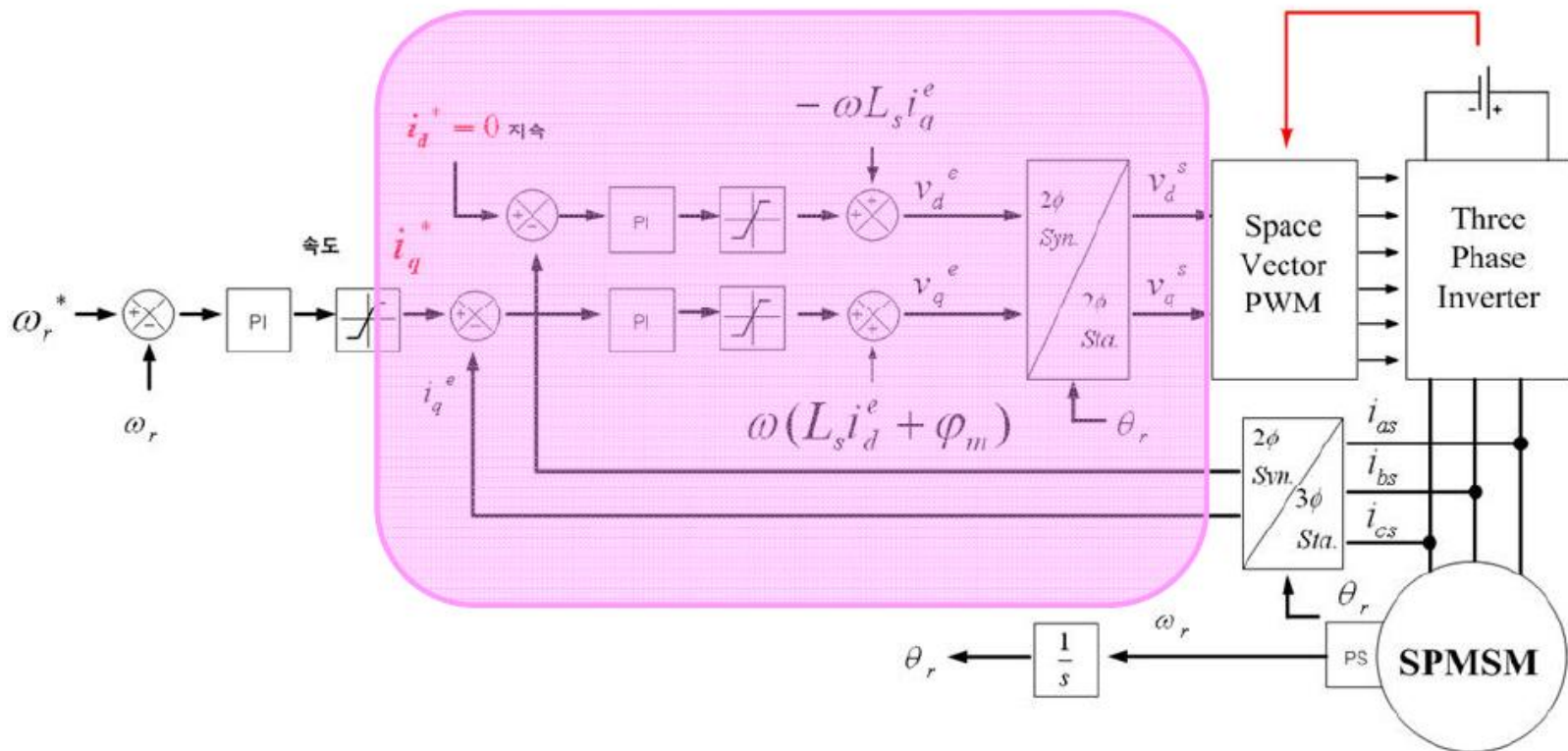
❖ 회전자 자속  $\phi_f$ 와 고정자 전류 **q**축 성분만 관여!!!

# Simulink를 이용한 vector 제어 model(2)

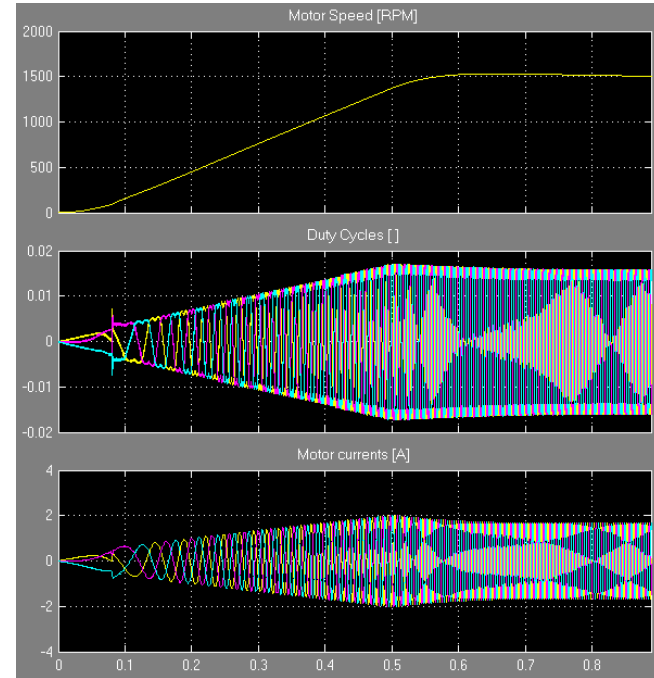
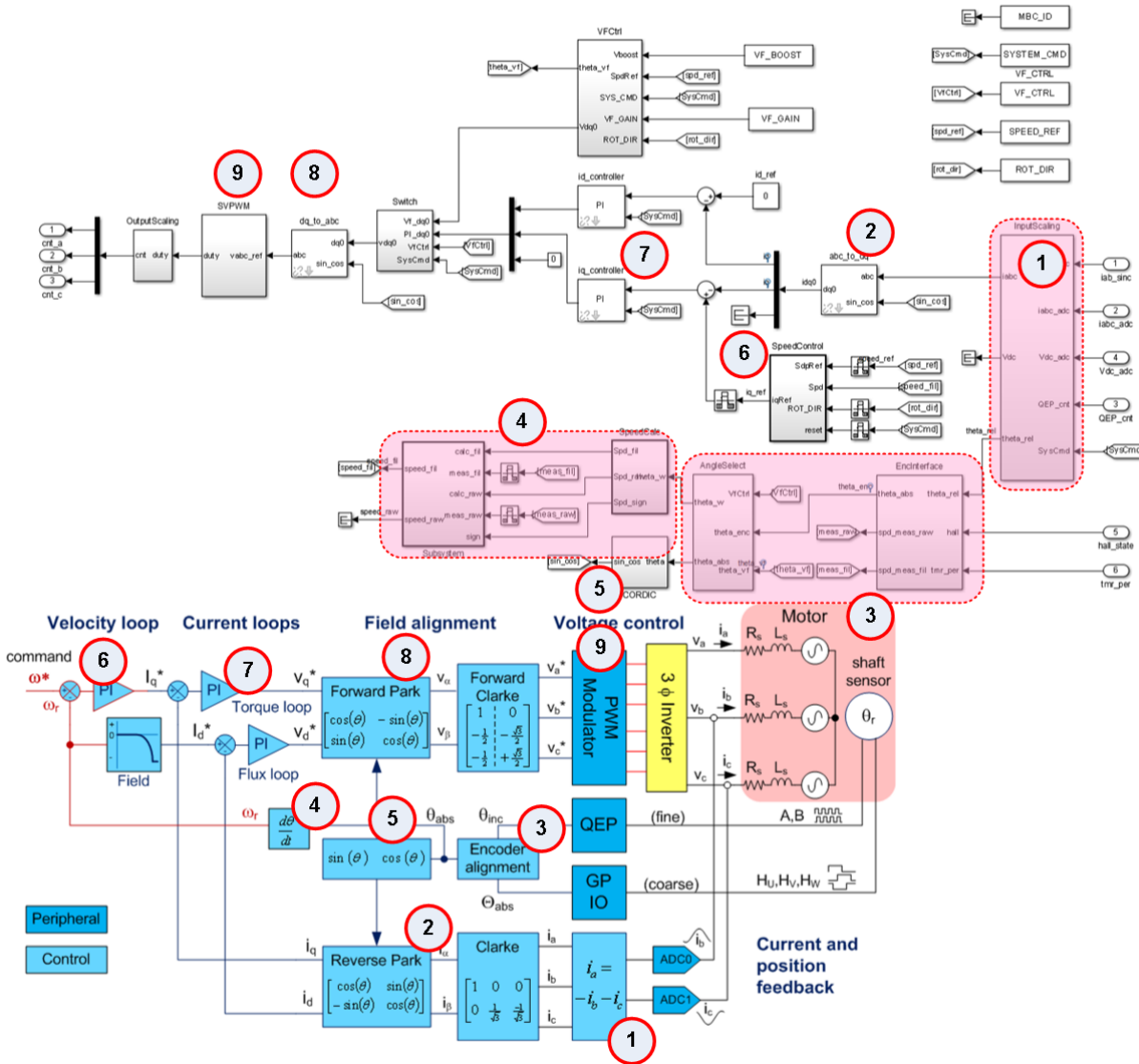
## ◆ SPMSM 분석 :

- q PI d-q

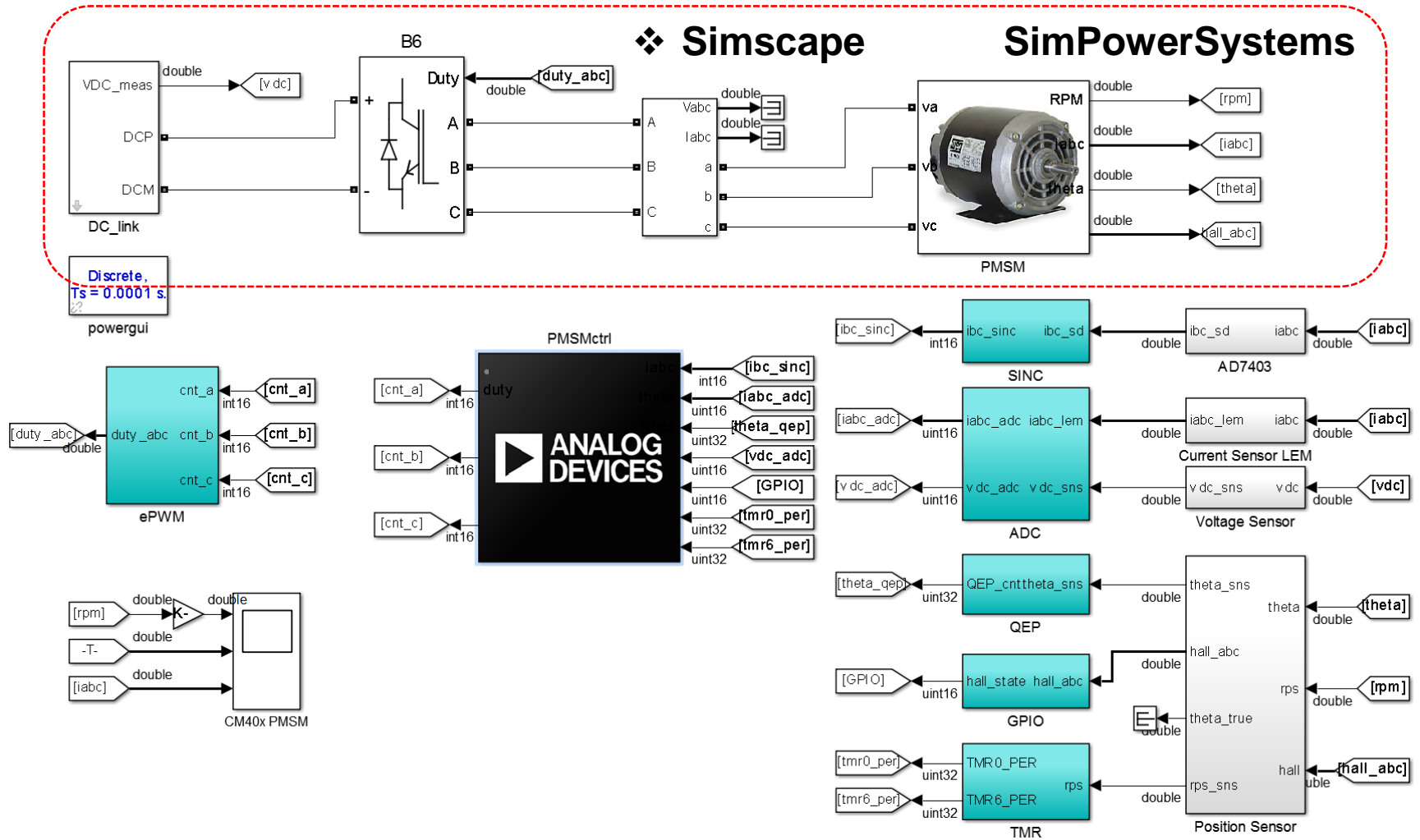
### 전류 제어기



# Simulink로 생성한 Code로 Motor 제어(1)



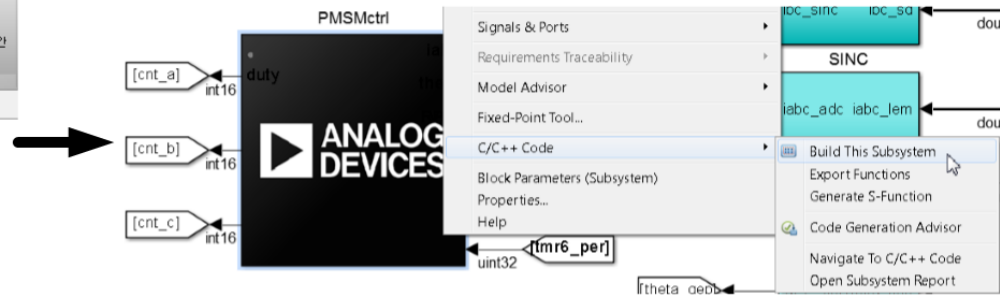
# Simulink로 생성한 Code로 Motor 제어(2)



# Simulink로 생성한 Code로 Motor 제어(3)

```

19 % October 11, 2012
20
21 %----- BEGIN CODE -----
22 clear all;
23
24 %Global enum that hold supported motor types
25 MOTOR = struct('ANAHEIM_24V', 0, 'TEKNIC_110V', 1, 'KOLLMORGEN_AKM22', 2);
26 %MOTOR_CFG = MOTOR.ANAHEIM_24V; % Select motor to be used
27 MOTOR_CFG = MOTOR.TEKNIC_110V; % Select motor to be used
    
```



**Code Generation Report for 'PMSMctrl'**

**Summary**

Code generation for model "PMSMctrl"

Model version	1.2421
Simulink Coder version	8.6 (R2014a) 27-Dec-2013
C source code generated on	Mon Oct 06 15:33:53 2014

Configuration settings at the time of code generation: [click to open](#)

Code generation objectives: **Unspecified**

Validation result: Not run

Buttons: OK, Help

**Build code for Subsystem:PMSMctrl**

Pick tunable parameters

Variable Name	Class	Storage Class
EncoderPulses	single	Inlined
GainMax	single	Inlined
GainMin	single	Inlined
Kdw	single	Inlined
...	...	...

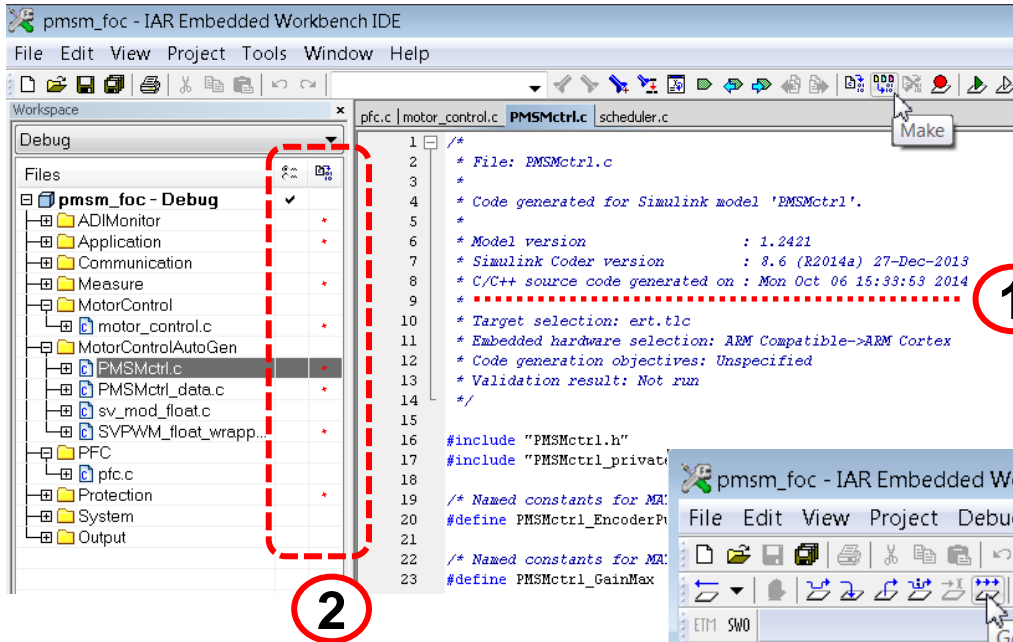
Blocks using selected variable

Block	Parent

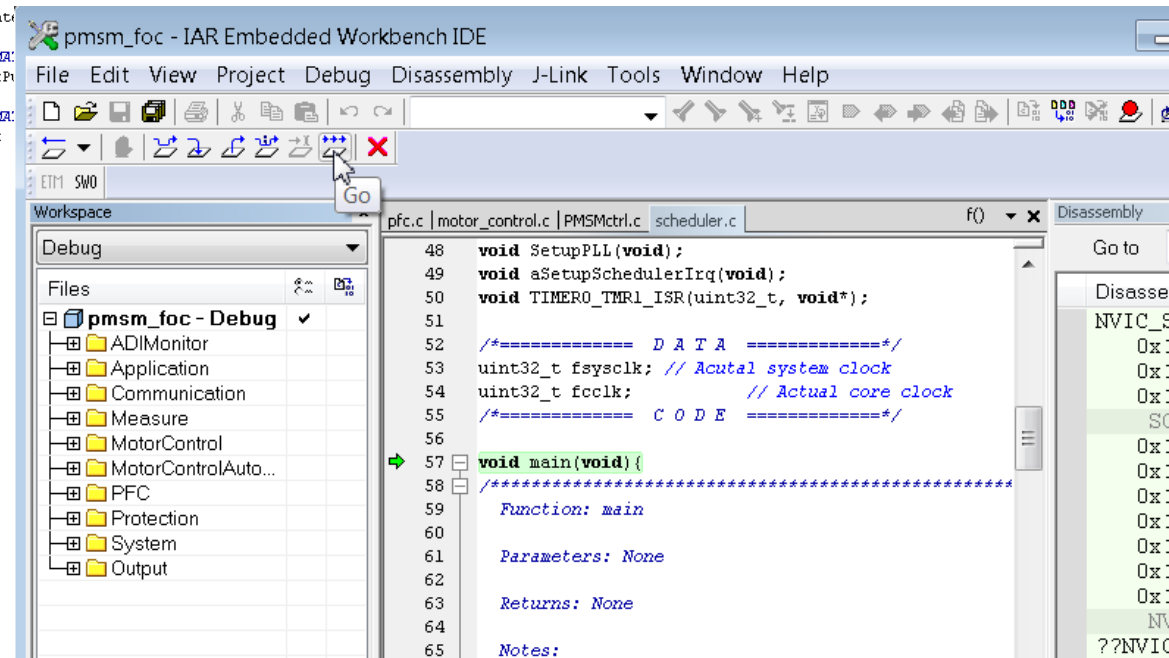
Buttons: Build, Cancel, Help

Status: Select tunable parameters and click Build

# Simulink로 생성한 Code로 Motor 제어(4)



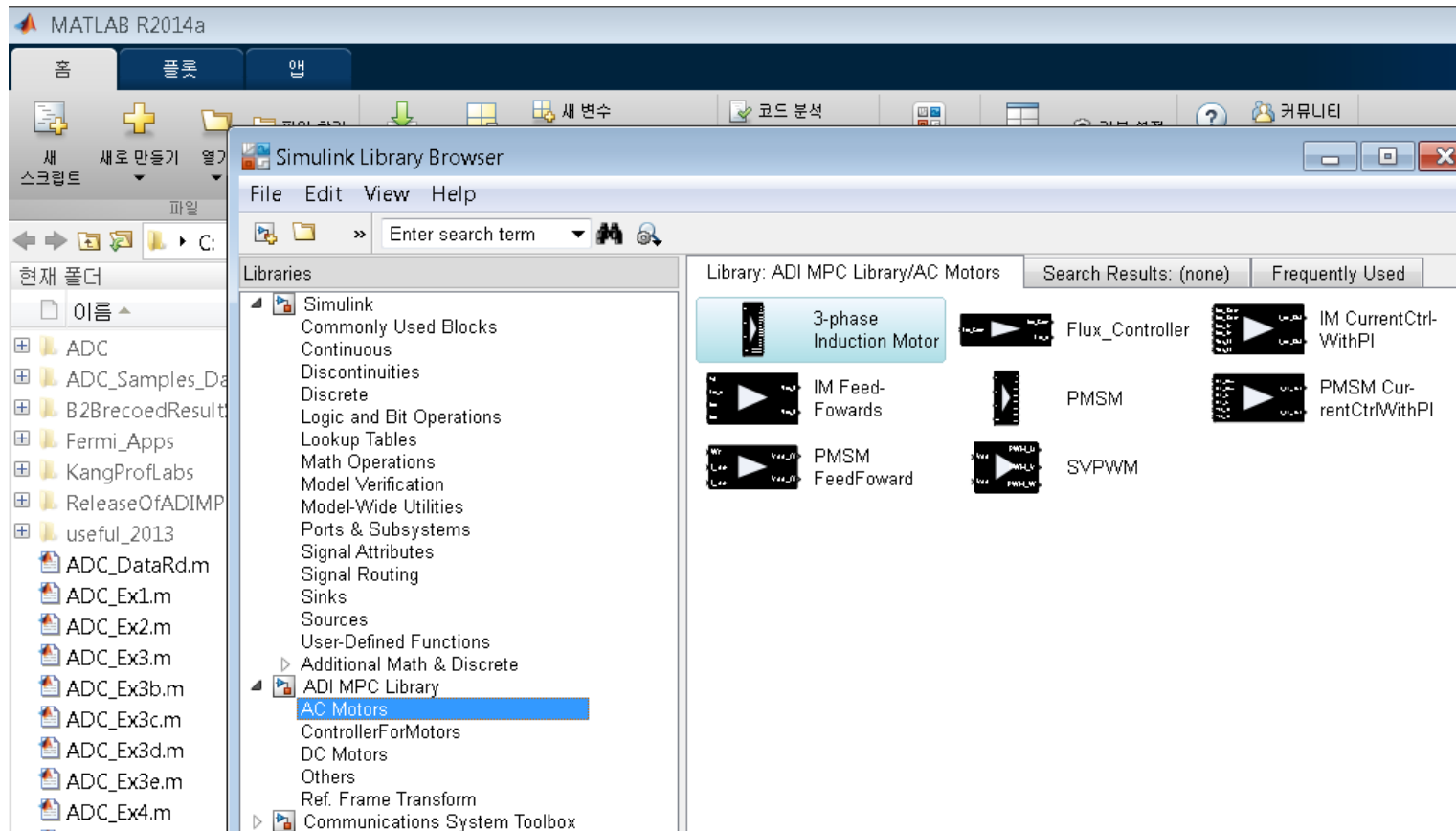
Download & Debug



# 기 타(1)

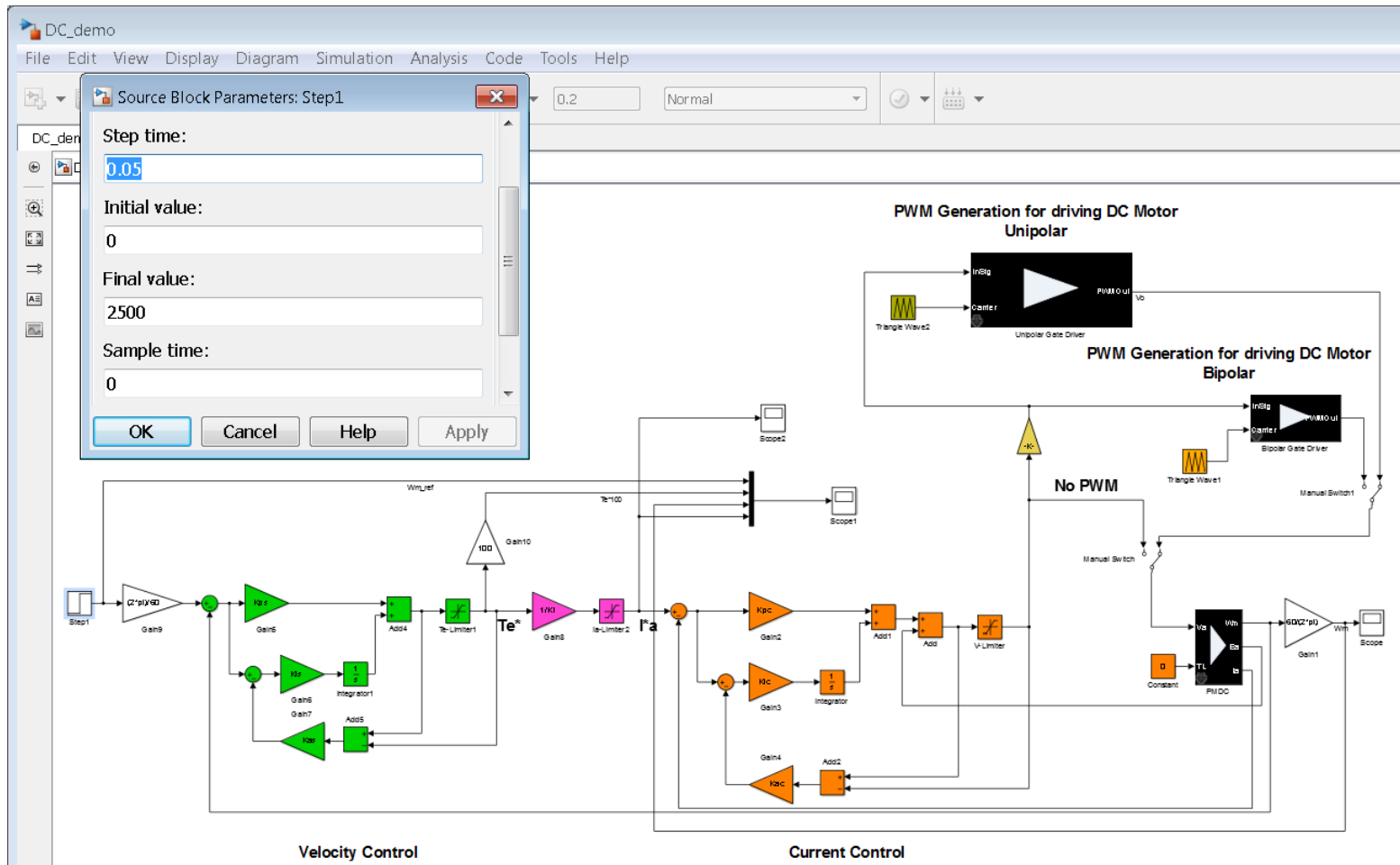
## ◆ 자신만의 Simulink Library 생성 :

- Masked block, s-function block !



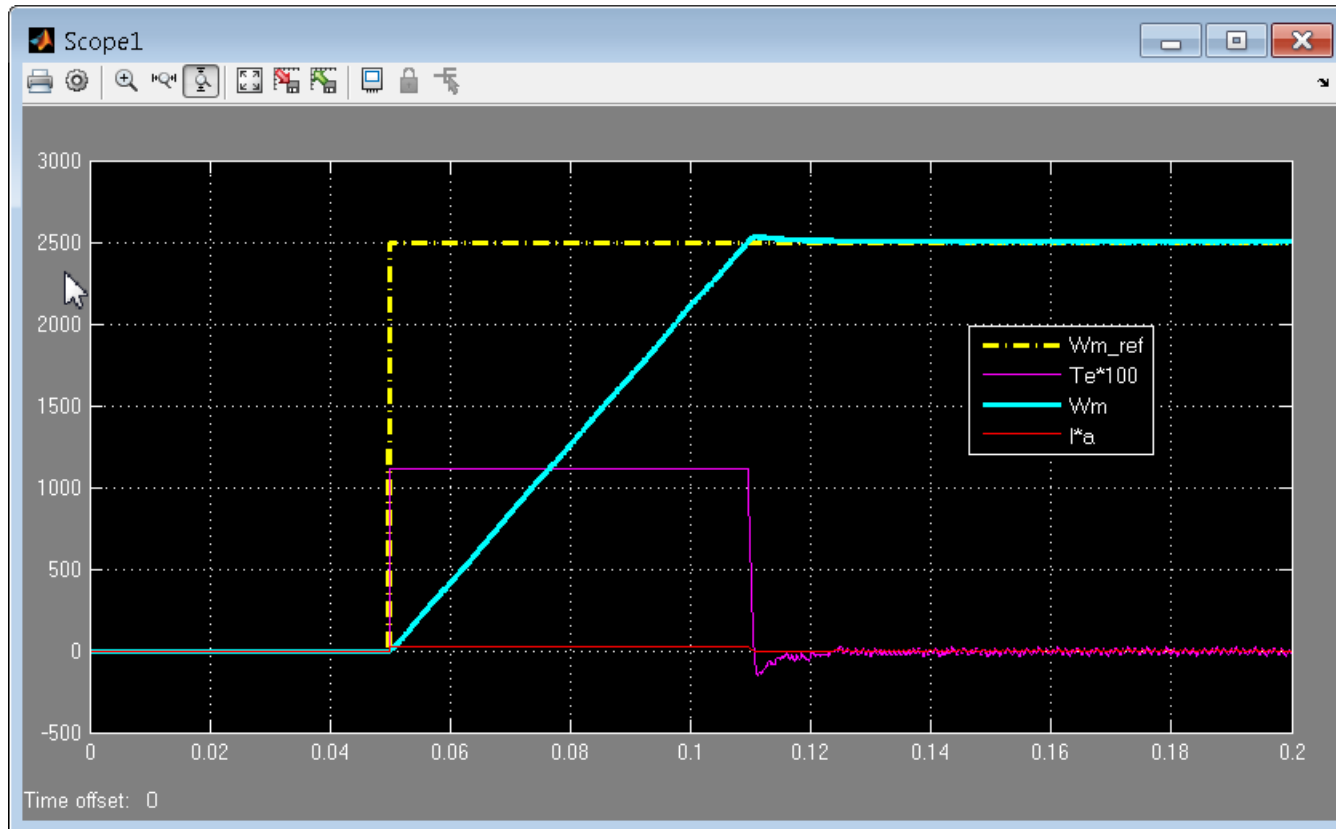
## 기 타(2)

- ◆ 생성 Library를 이용한 Model 생성 :



## 기 타(3)

- ◆ Simulation 검증 및 C-code 생성 :



## 참고 문헌

- ◆ “임종수의 전력 전자 강의”
  - : <http://cafe.naver.com/limjongsulab>
- ◆ 전기기기 제어론, 설승기 교수님. 홍릉과학 출판사.
- ◆ DC, AC, BLDC 모터 제어, 김상훈 교수님, 복두 출판사.
- ◆ 임종수의 Cortex-M3/M4 완벽 가이드(기초편), (응용편)