임베디드시스템 기초(#514115)

#10. Analog-Digital Converter 2/2

한림대학교
전자공학과 이선우

한림대학교
HALLYM UNIVERSITY

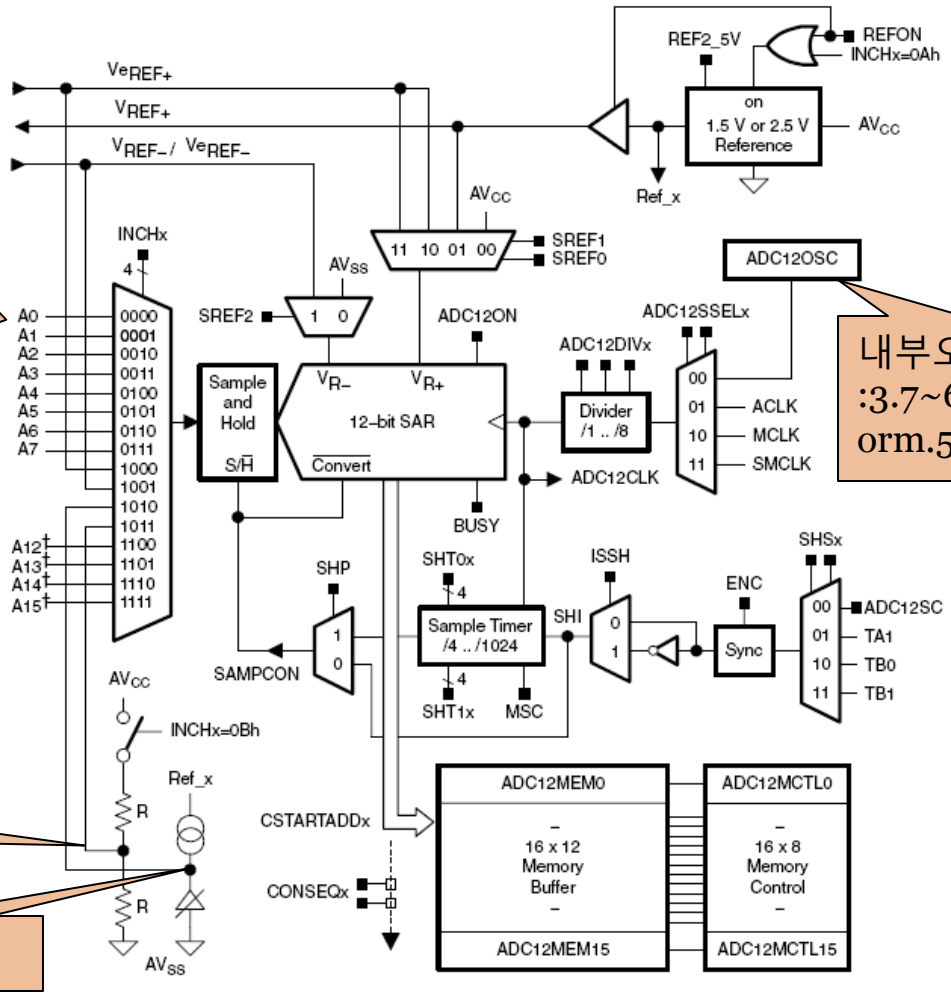
Contents

- MSP430 ADC12
  - Block diagram
  - Operation
  - Control registers
  - Example codes

ADC12 block diagram

Figure 26–1. ADC12 Block Diagram

8개의 아날로그 입력 채널(A0~A7)+4개의 내부 입력(VeRef+/-,Avcc/2, Temp. sensor)
*4개 추가 채널(A12~15, 43x/461x only)



내부오실레이터 :3.7~6.3MHz(norm.5MHz)

Avcc/2 입력

내부온도센서

† MSP430FG43x and MSP430FG461x devices only

ADC12 Operation

- 12-Bit ADC Core
  - ADC core converts an analog input to 12-bit digital code.
  - Transfer function: $N_{ADC} = 4095*(V_{IN} – V_{R-})/(V_{R+} - V_{R-})$
  - Conversion clock selection
    - 변환을 위해서는 필요한 클럭 신호
    - 4가지 소스(MCLK, SMCLK, SCLK, ADC12OSC) 중 하나 선택 사용.
- Inputs & Mux.
  - 8개의 외부(A0~A7), 4개의 내부 아날로그 입력 중 하나를 선택해서 변환.
  - 8개 외부 입력 사용 위해서는 P6SEL 선택해야 함.
- Ref. Voltage Generator
  - 기준 전압 발생기 내장: 1.2, 2.5V 선택 가능. 혹은 외부 기준전압(V_{REF+} pin 입력) 사용 가능

ADC12 Operation

- Auto Power-Down
  - 소비전력 감소를 위해 ADC12가 변환하지 않을 때 자동으로 꺼진다 (Core & ADC12OSC disabled).
  - 기준전압발생기는 수동으로 disable 시킴.(REFON=0)
- Sample & conversion Timing
  - Sampling: 입력 전압의 크기와 동일한 전압을 만들고 이를 고정시키는 작업(대개 capacitor 이용). 따라서 일정 시간이 필요.이를 sample period (t_{sample})라 함.
  - AD변환은 SHI(sample input signal)의 rising edge로 initiated(started)됨.
  - SHI의 소스: 4개 (ADC12SC, TA1, TB0, TB1)중 하나 선택
  - 2개 sample mode
    - Extended Sample Mode: SHI 신호의 T_{ON} 동안 sampling
    - Pulse Sample Mode: SHI는 triggering만 (t_{sample} 은 미리 지정)

Sample and Conversion Timing

Figure 26–3. Extended Sample Mode

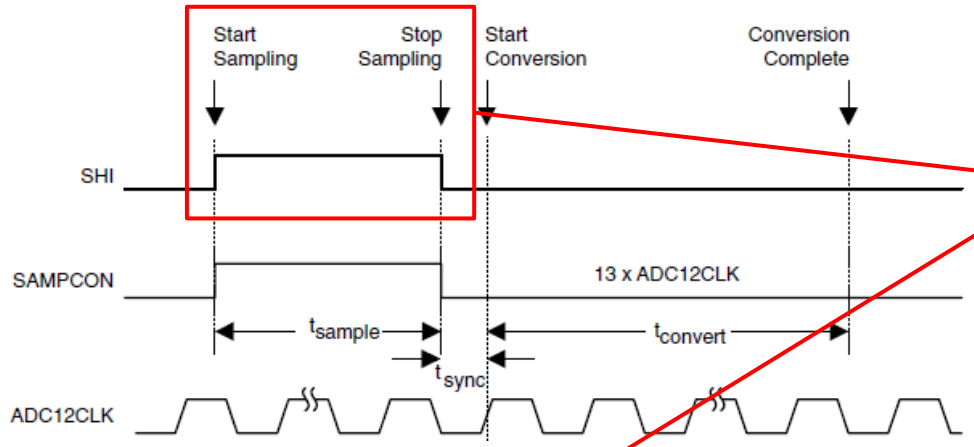
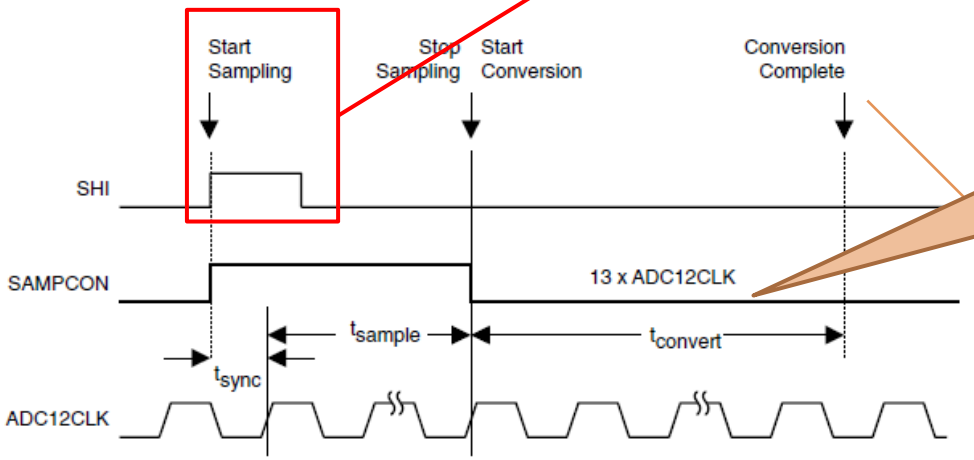


Figure 26–4. Pulse Sample Mode



2 Mode 차이점
•Sample mode: SHI가 H일 동안
• Pulse mode: SHI의 rising edge에 의해 start.

• Conversion time: 13개 ADC12CLK 펄스가 필요함.
• f가 크면 변환시간 짧음.

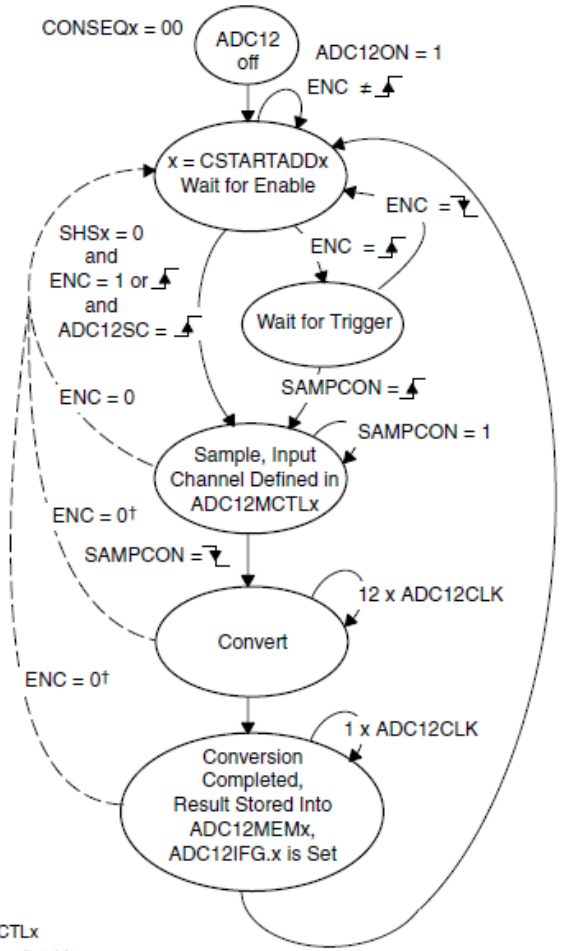
한림대학교 HALLYM UNIVERSITY

Conversion Memory & Modes

- Memory
  - 16개 ADC12MEMx: 각각의 ADC12MCTLx reg.에 의해 제어
  - 각 메모리는 입력 채널 및 기준전압을 설정함.
  - ADC12CTL1 reg.의 CSTARTADDx bit로 변환 후 결과를 어떤 메모리(xxMCTLx)부터 저장할지를 결정. 만약 single-channel, repeat-single-channel일 경우는 하나의 ADC12MCTLx만 사용됨.
- Conversion modes
  - CONSEQx bit로 4개 모드 중 하나 선택.
  - Single-channel single-conversion: 한 채널 한번만 변환
  - Sequence-of-channels: 일련의 채널들을 한번씩만 변환
  - Repeat-single-channel: 한 채널을 여러 번 변환
  - Repeat-sequence-of-channels: 일련의 채널들을 여러 번 변환

State diagram: single ch. single conv. mode

Figure 26–6. Single-Channel, Single-Conversion Mode



x = pointer to ADC12MCTLx
†Conversion result is unpredictable

ADC12CTL0, ADC12 Control Register 0

15	14	13	12	11	10	9	8
SHT1x				SHT0x			
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

7	6	5	4	3	2	1	0
MSC	REF2_5V	REFON	ADC12ON	ADC12OVIE	ADC12 TOVIE	ENC	ADC12SC
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

ADC12CTL1, ADC12 Control Register 1

15	14	13	12	11	10	9	8
CSTARTADDx				SHSx		SHP	ISSH
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

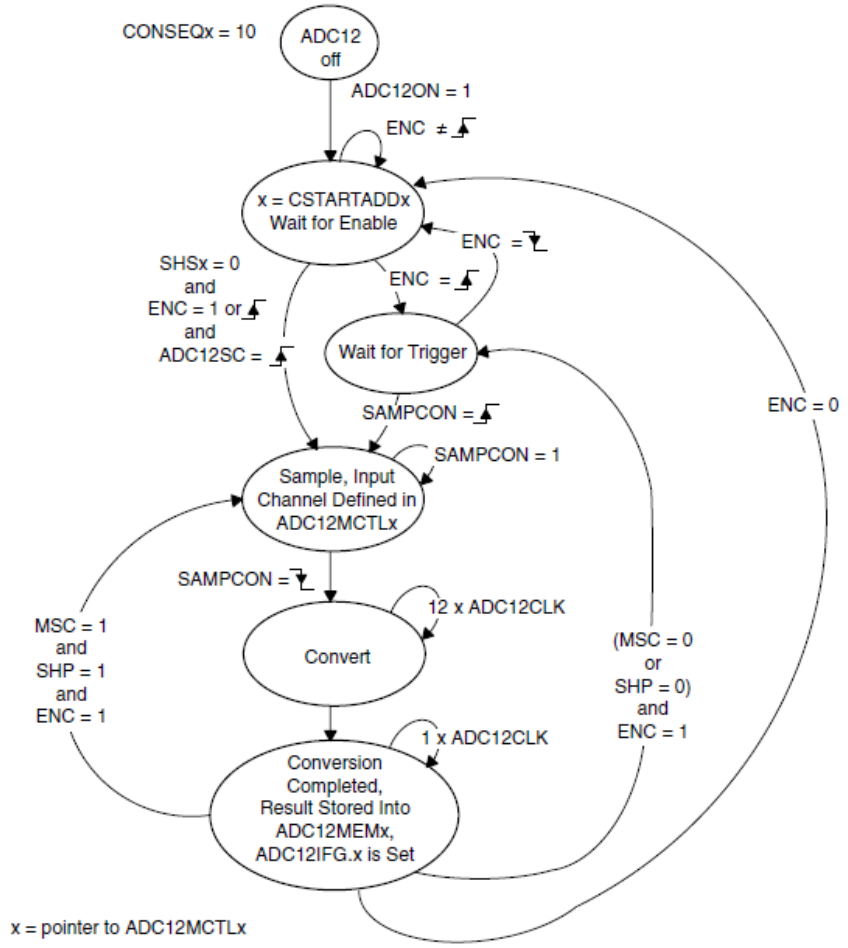
7	6	5	4	3	2	1	0
ADC12DIVx			ADC12SSELx		CONSEQx		ADC12 BUSY
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	r–(0)

ADC12MCTLx, ADC12 Conversion Memory Control Registers

7	6	5	4	3	2	1	0
EOS	SREFx			INCHx			
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

State diagram: repeat-single-channel mode

Figure 26–8. Repeat-Single-Channel Mode



ADC12CTL0, ADC12 Control Register 0

15	14	13	12	11	10	9	8
SHT1x				SHT0x			
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

7	6	5	4	3	2	1	0
MSC	REF2_5V	REFON	ADC12ON	ADC12OVIE	ADC12 TOVIE	ENC	ADC12SC
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

ADC12CTL1, ADC12 Control Register 1

15	14	13	12	11	10	9	8
CSTARTADDx				SHSx		SHP	ISSH
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

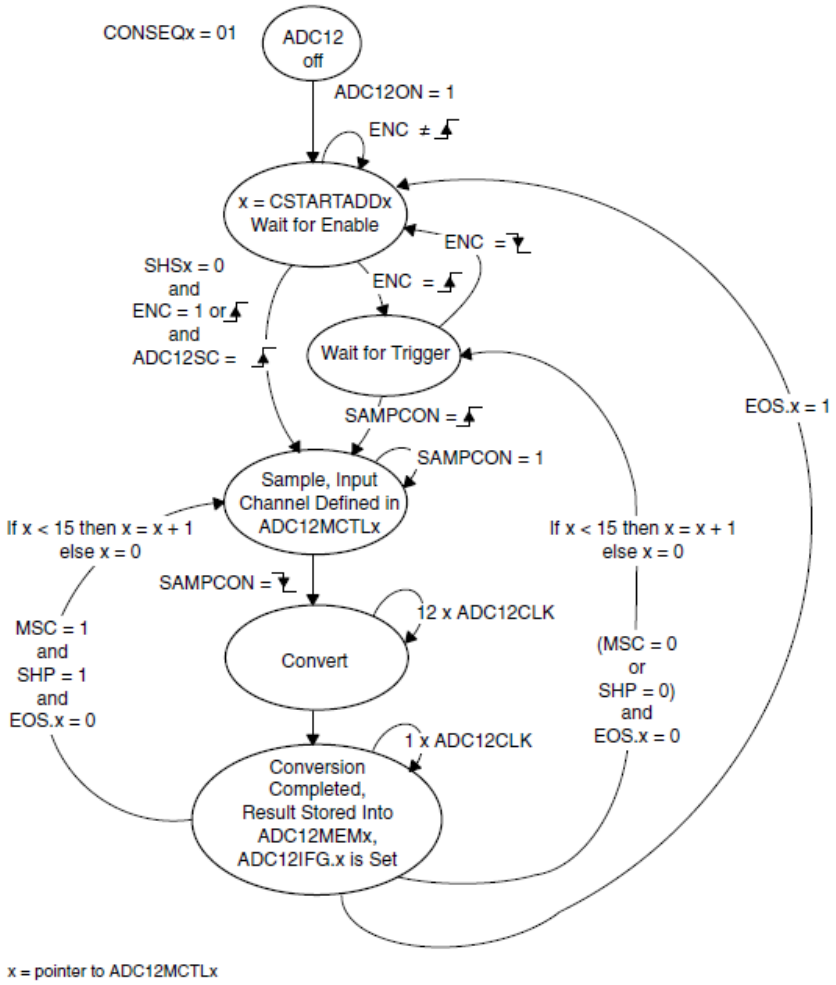
7	6	5	4	3	2	1	0
ADC12DIVx			ADC12SSELx		CONSEQx		ADC12 BUSY
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	r–(0)

ADC12MCTLx, ADC12 Conversion Memory Control Registers

7	6	5	4	3	2	1	0
EOS	SREFx			INCHx			
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

State diagram: sequence-of-channels mode

Figure 26–7. Sequence-of-Channels Mode



ADC12CTL0, ADC12 Control Register 0

15	14	13	12	11	10	9	8
SHT1x				SHT0x			
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

7	6	5	4	3	2	1	0
MSC	REF2_5V	REFON	ADC12ON	ADC12OVIE	ADC12 TOVIE	ENC	ADC12SC
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

ADC12CTL1, ADC12 Control Register 1

15	14	13	12	11	10	9	8
CSTARTADDx				SHSx		SHP	ISSH
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

7	6	5	4	3	2	1	0
ADC12DIVx			ADC12SSELx		CONSEQx		ADC12 BUSY
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	r–(0)

ADC12MCTLx, ADC12 Conversion Memory Control Registers

7	6	5	4	3	2	1	0
EOS	SREFx			INCHx			
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

ADC12CTL0: ADC12 control reg. 0

ADC12CTL0, ADC12 Control Register 0

15	14	13	12	11	10	9	8
SHT1x				SHT0x			
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

7	6	5	4	3	2	1	0
MSC	REF2_5V	REFON	ADC12ON	ADC12OVIE	ADC12 TOVIE	ENC	ADC12SC
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

Modifiable only when ENC = 0

SHT1x Bits 15-12 Sample-and-hold time. These bits define the number of ADC12CLK cycles in the sampling period for registers ADC12MEM8 to ADC12MEM15.

SHT0x Bits 11-8 Sample-and-hold time. These bits define the number of ADC12CLK cycles in the sampling period for registers ADC12MEM0 to ADC12MEM7.

SHTx Bits	ADC12CLK cycles
0000	4
0001	8
0010	16
0011	32
0100	64
0101	96
0110	128
0111	192
1000	256
1001	384
1010	512
1011	768
1100	1024
1101	1024
1110	1024
1111	1024

샘플링 시간 간격 설정. 즉, 내부 커패시터를 충전 시키는 시간임.
-너무 짧으면 입력 전압 보다 작은 전압 값으로 변환.
-너무 느리면 전체 변환 시간이 느려짐.
-16개 메모리를 2개 그룹(0~7,8~15)으로 나누어 결정함.

MSC Bit 7 Multiple sample and conversion. Valid only for sequence or repeated modes.
0 The sampling timer requires a rising edge of the SHI signal to trigger each sample-and-conversion.
1 The first rising edge of the SHI signal triggers the sampling timer, but further sample-and-conversions are performed automatically as soon as the prior conversion is completed.

REF2_5V Bit 6 Reference generator voltage. REFON must also be set.
0 1.5 V
1 2.5 V

REFON Bit 5 Reference generator on
0 Reference off
1 Reference on

ADC12ON Bit 4 ADC12 on
0 ADC12 off
1 ADC12 on

ADC12OVIE Bit 3 ADC12MEMx overflow-interrupt enable. The GIE bit must also be set to enable the interrupt.
0 Overflow interrupt disabled
1 Overflow interrupt enabled

ADC12 TOVIE Bit 2 ADC12 conversion-time-overflow interrupt enable. The GIE bit must also be set to enable the interrupt.
0 Conversion time overflow interrupt disabled
1 Conversion time overflow interrupt enabled

ENC Bit 1 Enable conversion
0 ADC12 disabled
1 ADC12 enabled

ADC12SC Bit 0 Start conversion. Software-controlled sample-and-conversion start. ADC12SC and ENC may be set together with one instruction. ADC12SC is reset automatically.
0 No sample-and-conversion-start
1 Start sample-and-conversion

ADC12CTL1: ADC12 control reg. 1

ADC12CTL1, ADC12 Control Register 1

15	14	13	12	11	10	9	8
CSTARTADDx				SHSx		SHP	ISSH
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

7	6	5	4	3	2	1	0
ADC12DIVx			ADC12SSELx		CONSEQx		ADC12 BUSY
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	r–(0)

Modifiable only when ENC = 0

CSTARTADDx Bits 15-12 Conversion start address. These bits select which ADC12 conversion-memory register is used for a single conversion or for the first conversion in a sequence. The value of CSTARTADDx is 0 to 0Fh, corresponding to ADC12MEM0 to ADC12MEM15.

SHSx Bits 11-10 Sample-and-hold source select
- 00 ADC12SC bit
- 01 Timer_A.OUT1
- 10 Timer_B.OUT0
- 11 Timer_B.OUT1

SHP Bit 9 Sample-and-hold pulse-mode select. This bit selects the source of the sampling signal (SAMPCON) to be either the output of the sampling timer or the sample-input signal directly.
- 0 SAMPCON signal is sourced from the sample-input signal.
- 1 SAMPCON signal is sourced from the sampling timer.

ISSH Bit 8 Invert signal sample-and-hold
- 0 The sample-input signal is not inverted.
- 1 The sample-input signal is inverted.

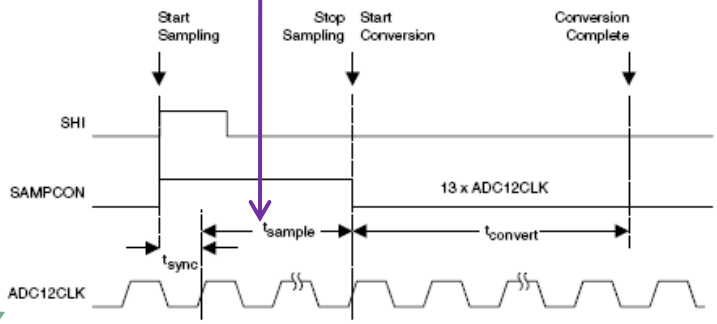
ADC12DIVx Bits 7-5 ADC12 clock divider
- 000 /1
- 001 /2
- 010 /3
- 011 /4
- 100 /5
- 101 /6
- 110 /7
- 111 /8

ADC12CLK 주파수 결정

16개 메모리중 어떤 것을 사용할지를 결정!
<u>싱글채널 단일변환</u> 시는 대개 메모리 하나만 사용!

ADC12CTL0의 SHT0/1에 따라 이 간격 결정!

Figure 26–4. Pulse Sample Mode



ADC12SSELx Bits 4-3 ADC12 clock source select
- 00 ADC12OSC
- 01 ACLK
- 10 MCLK
- 11 SMCLK

CONSEQx Bits 2-1 Conversion sequence mode select
- 00 Single-channel, single-conversion
- 01 Sequence-of-channels
- 10 Repeat-single-channel
- 11 Repeat-sequence-of-channels

ADC12BUSY Bit 0 ADC12 busy. This bit indicates an active sample or conversion operation.
- 0 No operation is active.
- 1 A sequence, sample, or conversion is active.

변환방법 설정!
<u>싱글채널 단일변환만</u> 공부함.
(CONSEQ:00)

변환 완료 상태

ADC12MCTLx: memory control reg.

ADC12MCTLx, ADC12 Conversion Memory Control Registers

7	6	5	4	3	2	1	0
EOS	SREFx			INCHx			
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

Modifiable only when ENC = 0

EOS Bit 7 End of sequence. Indicates the last conversion in a sequence.
- 0 Not end of sequence
- 1 End of sequence

SREFx Bits 6-4 Select reference
- 000 $V_{R+} = AV_{CC}$ and $V_{R-} = AV_{SS}$
- 001 $V_{R+} = V_{REF+}$ and $V_{R-} = AV_{SS}$
- 010 $V_{R+} = Ve_{REF+}$ and $V_{R-} = AV_{SS}$
- 011 $V_{R+} = Ve_{REF+}$ and $V_{R-} = AV_{SS}$
- 100 $V_{R+} = AV_{CC}$ and $V_{R-} = V_{REF-}/ Ve_{REF-}$
- 101 $V_{R+} = V_{REF+}$ and $V_{R-} = V_{REF-}/ Ve_{REF-}$
- 110 $V_{R+} = Ve_{REF+}$ and $V_{R-} = V_{REF-}/ Ve_{REF-}$
- 111 $V_{R+} = Ve_{REF+}$ and $V_{R-} = V_{REF-}/ Ve_{REF-}$

INCHx Bits 3-0 Input channel select
- 0000 A0
- 0001 A1
- 0010 A2
- 0011 A3
- 0100 A4
- 0101 A5
- 0110 A6
- 0111 A7
- 1000 Ve_{REF+}
- 1001 V_{REF-}/Ve_{REF-}
- 1010 Temperature sensor
- 1011 $(AV_{CC} – AV_{SS}) / 2$
- 1100 $(AV_{CC} – AV_{SS}) / 2$, A12 on 'FG43x and 'FG461x devices
- 1101 $(AV_{CC} – AV_{SS}) / 2$, A13 on 'FG43x and 'FG461x devices
- 1110 $(AV_{CC} – AV_{SS}) / 2$, A14 on 'FG43x and 'FG461x devices
- 1111 $(AV_{CC} – AV_{SS}) / 2$, A15 on 'FG43x and 'FG461x devices

MSP430 ADC의 구별/특징
- 16개 메모리 각각에 대해 각자 기준전압 및 입력채널을 설정할 수 있다.
- 이는 4개 동작 모드(싱글채널–단일/반복 변환, 여러 채널–단일/반복 변환) 제공 가능하게 함.

ADC12MEMx, ADC12 Conversion Memory Registers

15	14	13	12	11	10	9	8
0	0	0	0	Conversion Results			
r0	r0	r0	r0	rw	rw	rw	rw

7	6	5	4	3	2	1	0
Conversion Results							
rw	rw	rw	rw	rw	rw	rw	rw

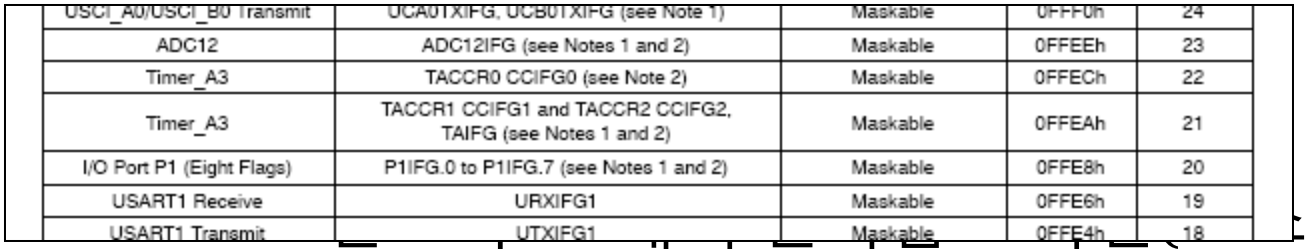
Conversion Results Bits 15-0 The 12-bit conversion results are right-justified. Bit 11 is the MSB. Bits 15-12 are always 0. Writing to the conversion memory registers will corrupt the results.

Example code: Sing.-ch. Sing. Conv. (polling)

```
void main(void)
{
//전략…
  P6SEL |= 0x01; //enable A0 (P6.0)
  ADC12CTL0 = REFON + REF2_5V+SHT0_2 + ADC12ON; //turn on
  2.5V, S&H time=16 cycles
  ADC12CTL1 = SHP;  //sampling timer 사용, CSTARTADDx=0
  ADC12MCTL0 = SREF_1; //Vr+=Vref+

  for(i=0x3600;I;i--); //delay for needed Vref start-up

  ADC12CTL0 |= ENC;

  while(1) {
     ADC12CTL0 |= ADC12SC;  //start conversion
     while( !(ADC12IFG & 0x0001) ); //conversion done?
     //read result value
     ADResult = ADC12MEM0;
  }
}
```

ADC12 관련 인터럽트

- MSP430xG461x 시리즈는 ADC12 관련 maskable irq. vector를 가짐.



USCI_A0/USCI_B0 Transmit	UCA0TXIFG, UCB0TXIFG (see Note 1)	Maskable	0FFF0h	24
ADC12	ADC12IFG (see Notes 1 and 2)	Maskable	0FFEEh	23
Timer_A3	TACCR0 CCIFG0 (see Note 2)	Maskable	0FFECh	22
Timer_A3	TACCR1 CCIFG1 and TACCR2 CCIFG2, TAIFG (see Notes 1 and 2)	Maskable	0FFEAh	21
I/O Port P1 (Eight Flags)	P1IFG.0 to P1IFG.7 (see Notes 1 and 2)	Maskable	0FFE8h	20
USART1 Receive	URXIFG1	Maskable	0FFE6h	19
USART1 Transmit	UTXIFG1	Maskable	0FFE4h	18

- 여기서 [illegible])들의 합성(ORed) 결과임. (즉 P1/2와 동일)
  - ADC12IFG0 ~ ADC12IFG15
  - ADC12OV: ADC12MEMx overflow, xMEMx 결과를 읽기 전에 다시 쓸 경우 발생.
  - ADC12TOV: Conversion Time Overflow, 현재 변환 중인데 새로운 변환을 시작하는 경우 발생
- 18개 irq. 소스도 다시 우선순위를 가짐
  - ADC12MEMx overflow: highest
  - ADC12MEM15 IFG: lowest

Irq. 관련 SFR:ADC12IV, ADC12IE/IFG

ADC12IV, ADC12 Interrupt Vector Register

15	14	13	12	11	10	9	8
0	0	0	0	0	0	0	0
r0	r0	r0	r0	r0	r0	r0	r0

7	6	5	4	3	2	1	0
0	0	ADC12IVx					0
r0	r0	r–(0)	r–(0)	r–(0)	r–(0)	r–(0)	r0

ADC12IVx Bits 15-0 ADC12 interrupt vector value

ADC12IV Contents	Interrupt Source	Interrupt Flag	Interrupt Priority
000h	No interrupt pending	–	
002h	ADC12MEMx overflow	–	Highest
004h	Conversion time overflow	–	
006h	ADC12MEM0 interrupt flag	ADC12IFG0	
008h	ADC12MEM1 interrupt flag	ADC12IFG1	
00Ah	ADC12MEM2 interrupt flag	ADC12IFG2	
00Ch	ADC12MEM3 interrupt flag	ADC12IFG3	
00Eh	ADC12MEM4 interrupt flag	ADC12IFG4	
010h	ADC12MEM5 interrupt flag	ADC12IFG5	
012h	ADC12MEM6 interrupt flag	ADC12IFG6	
014h	ADC12MEM7 interrupt flag	ADC12IFG7	
016h	ADC12MEM8 interrupt flag	ADC12IFG8	
018h	ADC12MEM9 interrupt flag	ADC12IFG9	
01Ah	ADC12MEM10 interrupt flag	ADC12IFG10	
01Ch	ADC12MEM11 interrupt flag	ADC12IFG11	
01Eh	ADC12MEM12 interrupt flag	ADC12IFG12	
020h	ADC12MEM13 interrupt flag	ADC12IFG13	
022h	ADC12MEM14 interrupt flag	ADC12IFG14	
024h	ADC12MEM15 interrupt flag	ADC12IFG15	Lowest

ADC12IE, ADC12 Interrupt Enable Register

15	14	13	12	11	10	9	8
ADC12IE15	ADC12IE14	ADC12IE13	ADC12IE12	ADC12IE11	ADC12IE10	ADC12IE9	ADC12IE8
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

7	6	5	4	3	2	1	0
ADC12IE7	ADC12IE6	ADC12IE5	ADC12IE4	ADC12IE3	ADC12IE2	ADC12IE1	ADC12IE0
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

ADC12IEx Bits 15-0 Interrupt enable. These bits enable or disable the interrupt request for the ADC12IFGx bits.
0 Interrupt disabled
1 Interrupt enabled

ADC12IFG, ADC12 Interrupt Flag Register

15	14	13	12	11	10	9	8
ADC12 IFG15	ADC12 IFG14	ADC12 IFG13	ADC12 IFG12	ADC12 IFG11	ADC12 IFG10	ADC12 IFG9	ADC12 IFG8
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

7	6	5	4	3	2	1	0
ADC12 IFG7	ADC12 IFG6	ADC12 IFG5	ADC12 IFG4	ADC12 IFG3	ADC12 IFG2	ADC12 IFG1	ADC12 IFG0
rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)	rw–(0)

ADC12IFGx Bits 15-0 ADC12MEMx Interrupt flag. These bits are set when corresponding ADC12MEMx is loaded with a conversion result. The ADC12IFGx bits are reset if the corresponding ADC12MEMx is accessed, or may be reset with software.
0 No interrupt pending
1 Interrupt pending

Example code: Sing.-ch. Sing. Conv. (use IRQ)

```
unsigned int ADResult;
void main(void)
{
//전략…
  P6SEL |= 0x01; //enable A0 (P6.0)
  ADC12CTL0 = REFON + REF2_5V+SHT0_15 + ADC12ON; //turn on 2.5V,
   S&H time=1024cycles(longer)
  ADC12CTL1 = SHP;  //sampling timer 사용, CSTARTADDx=0
  ADC12IE |= 0x01; //enable ADC12IFG.0
  ADC12MCTL0 = SREF_1; //Vr+=Vref+
  ADC12CTL0 |= ENC;
  __enable_interrupt();
  while(1) {
     ADC12CTL0 |= ADC12SC;  //start conversion
      _low_power_mode_0();
     _no_operation();
  }
}
#pragma vector=ADC12_VECTOR
__interrupt void ADC12_ISR(void)
{
   ADResult = ADC12MEM0;
    _low_power_mode_off_on_exit();
}
```

Example code: repeat-single-channel

```
unsigned int Result[8], index;
void main(void)
{
//전략…
  P6SEL |= 0x01; //enable A0 (P6.0)
  ADC12CTL0 = REFON + REF2_5V + SHT0_8 + ADC12ON + MSC;
  //turn on 2.5V, S&H time=256 cycles, multiple sampling
  ADC12CTL1 = SHP + CONSEQ_2;  //set mode=2
  ADC12IE |= 0x01; //enable ADC12IFG.0
  ADC12MCTL0 = SREF_1; //Vr+=Vref+
  ADC12CTL0 |= ENC;
  __enable_interrupt();
  ADC12CTL0 |= ADC12SC;  //start conversion
}

#pragma vector=ADC12_VECTOR
__interrupt void ADC12_ISR(void)
{
   Result[index++] = ADC12MEM0;
  index & = 0x07;
}
```