12주차

2016. 11. 16.

목차

- Atmega128의 주요 특징
- Timer / Counter
  : 8-bit Timer/Counter
  : 16-bit Timer/Counter
- Timer / Counter 실습
  : LED
  : PWM
- 레포트

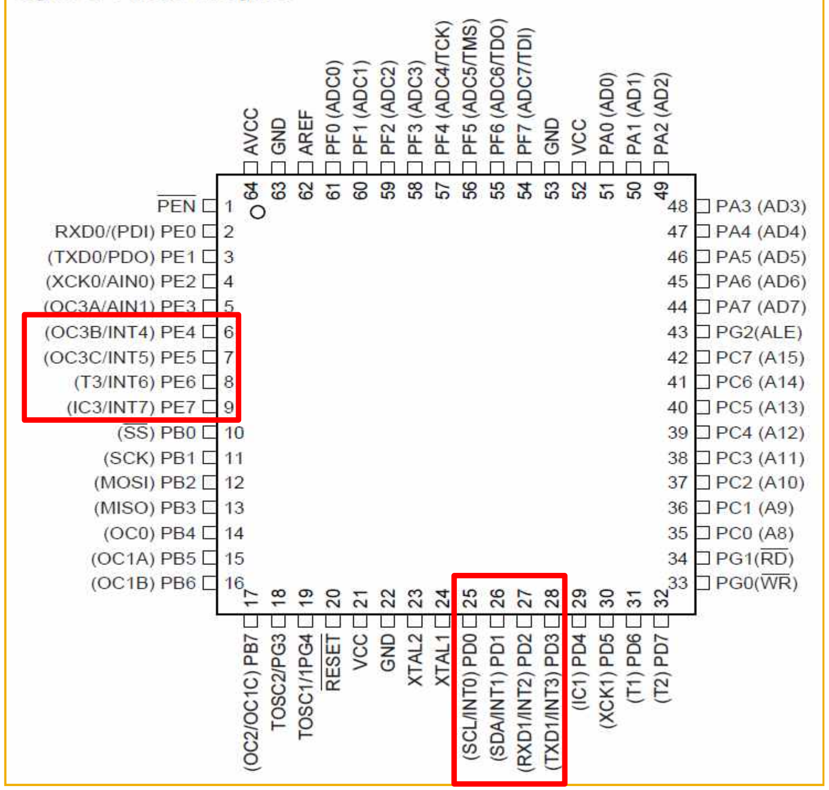
후반부 강의 계획

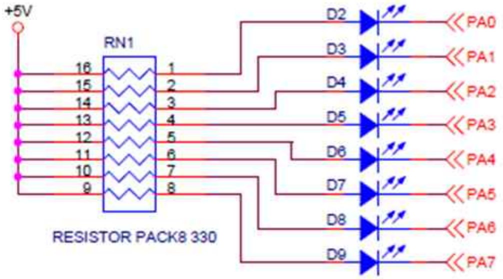
9	마이크로프로세서 개요1 C언어	C언어 강의	보드 납땜 (컨트롤러부, LED)
10	마이크로프로세서 개요2 GPIO 1	마이크로프로세서란? Atmega128의 구조와 기능 Firmware ? GPIO ? LED 제어	C언어 Quiz
11	Interrupt GPIO 2	Interrupt 이해 및 실습	C언어 Quiz
12	타이머/카운터	타이머/카운터 이해 및 실습 + LED	C언어 Quiz
13	주변장치 제어1	인터럽트와 4x4 keypad 제어	4x4 Keypad
14	주변장치 제어2	인터럽트와 타이머/카운터를 이용한 7-Segment 제어	7-Segment
15	주변장치 제어3	디지털 시계 / 스톱워치	
16	기말고사	기말고사 (이론 + 실기)	

Chapter 4 - 예제 1

```
#include <iom128.h>
#include <intrinsics.h>

void main(void)
{
   unsigned char number = 0xFE;

   DDRA = 0xFF;
   PORTA = 0xFF;

   DDRD = 0x0F;
   PORTD = 0x1F;

   while(1)
   {
    while(PIND & 0x10);
    while(!(PIND & 0x10));

    PORTA = number;
    number = (number << 1) | 0x01;

    if((number & 0xFF) == 0xFF)
       number =0xFE;

   }
}
```

Figure 1. Pinout ATmega128





교재
181 페이지
참고

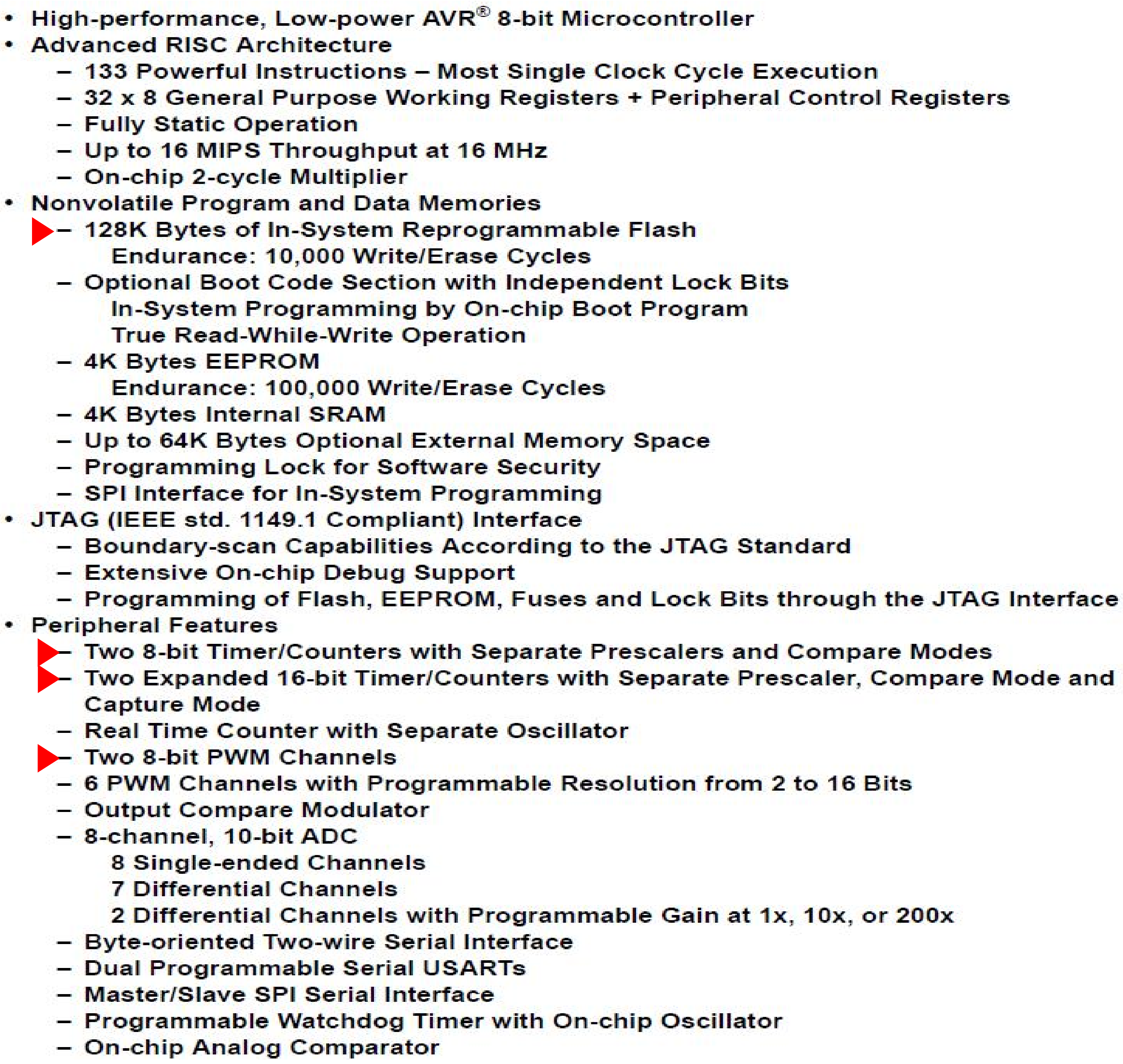
Chapter 5 - 예제 4

```
#include <iom128.h>
#include <intrinsics.h>

unsigned int data = 0xFE;

void main(void)
{
  DDRA = 0xFF;
  PORTA = 0xFF;

  DDRD = 0x0F;
  PORTD = 0x3F;

  EIMSK |= 0x30;  // INT4 and INT5 interrupt enable
  EICRB |= 0x0A;  // INT4 and INT5 falling edge
  EIFR |= 0x30;   // INT4 and INT5 interrupt flag clear

  __enable_interrupt();

  while(1);
}
```

```
#pragma vector=INT4_vect
__interrupt void INT4_interrupt(void)
{
  __disable_interrupt();

  PORTA = data;
  data--;

  __enable_interrupt();
}

#pragma vector=INT5_vect
__interrupt void INT5_interrupt(void)
{
  __disable_interrupt();

  PORTA = data;
  data++;

  __enable_interrupt();
}
```

교재
208 페이지
참고

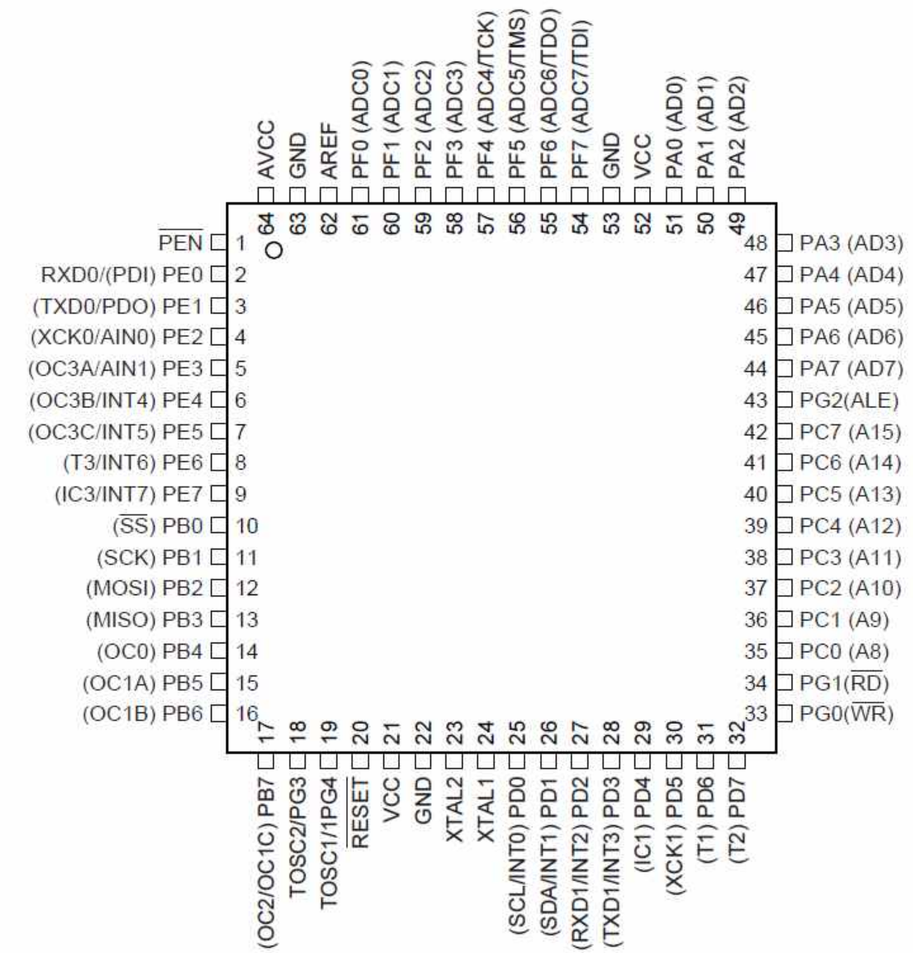
ATMEGA128의 주요 특징



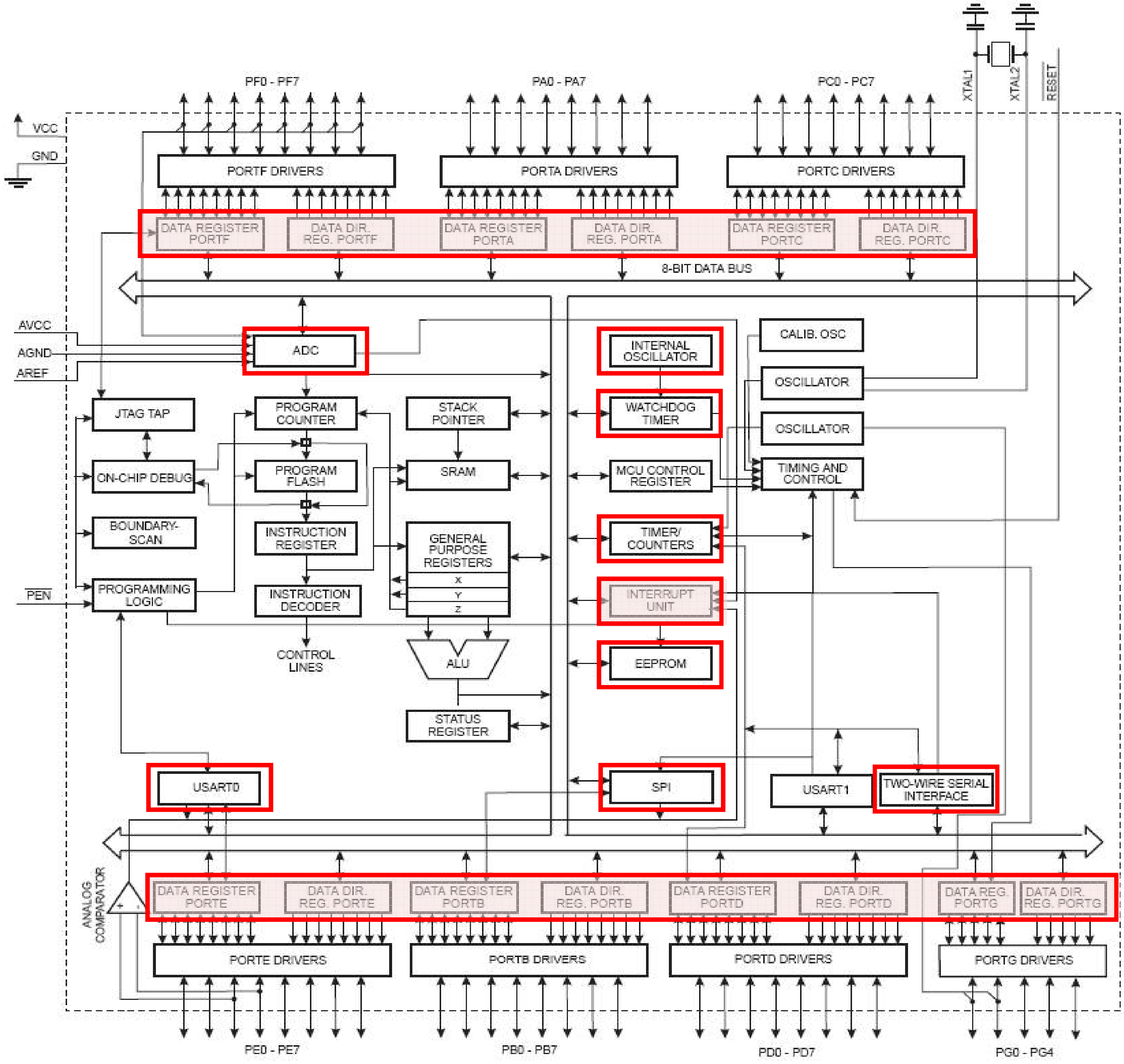
- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 133 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers + Peripheral Control Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
  - ▶ 128K Bytes of In-System Reprogrammable Flash
    Endurance: 10,000 Write/Erase Cycles
  - Optional Boot Code Section with Independent Lock Bits
    In-System Programming by On-chip Boot Program
    True Read-While-Write Operation
  - 4K Bytes EEPROM
    Endurance: 100,000 Write/Erase Cycles
  - 4K Bytes Internal SRAM
  - Up to 64K Bytes Optional External Memory Space
  - Programming Lock for Software Security
  - SPI Interface for In-System Programming
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses and Lock Bits through the JTAG Interface
- Peripheral Features
  - ▶ Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
  - ▶ Two Expanded 16-bit Timer/Counters with Separate Prescaler, Compare Mode and Capture Mode
  - Real Time Counter with Separate Oscillator
  - ▶ Two 8-bit PWM Channels
  - 6 PWM Channels with Programmable Resolution from 2 to 16 Bits
  - Output Compare Modulator
  - 8-channel, 10-bit ADC
    8 Single-ended Channels
    7 Differential Channels
    2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
  - Byte-oriented Two-wire Serial Interface
  - Dual Programmable Serial USARTs
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with On-chip Oscillator
  - On-chip Analog Comparator

- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
  - Software Selectable Clock Frequency
  - ATmega103 Compatibility Mode Selected by a Fuse
  - Global Pull-up Disable
- I/O and Packages
  - 53 Programmable I/O Lines
  - 64-lead TQFP and 64-pad MLF
- Operating Voltages
  - 2.7 - 5.5V for ATmega128L
  - 4.5 - 5.5V for ATmega128
- Speed Grades
  - 0 - 8 MHz for ATmega128L
  - 0 - 16 MHz for ATmega128

Figure 1. Pinout ATmega128



교재 15-20 페이지 참고


PF0 - PF7
PA0 - PA7
PC0 - PC7
XTAL1
XTAL2
RESET
VCC
GND
PORTF DRIVERS
PORTA DRIVERS
PORTC DRIVERS
DATA REGISTER PORTF
DATA DIR. REG. PORTF
DATA REGISTER PORTA
DATA DIR. REG. PORTA
DATA REGISTER PORTC
DATA DIR. REG. PORTC
8-BIT DATA BUS
AVCC
AGND
AREF
ADC
INTERNAL OSCILLATOR
CALIB. OSC
OSCILLATOR
OSCILLATOR
JTAG TAP
PROGRAM COUNTER
STACK POINTER
WATCHDOG TIMER
ON-CHIP DEBUG
PROGRAM FLASH
SRAM
MCU CONTROL REGISTER
TIMING AND CONTROL
BOUNDARY-SCAN
INSTRUCTION REGISTER
GENERAL PURPOSE REGISTERS
X
Y
Z
TIMER/ COUNTERS
PEN
PROGRAMMING LOGIC
INSTRUCTION DECODER
INTERRUPT UNIT
CONTROL LINES
ALU
EEPROM
STATUS REGISTER
USART0
SPI
USART1
TWO-WIRE SERIAL INTERFACE
ANALOG COMPARATOR
DATA REGISTER PORTE
DATA DIR. REG. PORTE
DATA REGISTER PORTB
DATA DIR. REG. PORTB
DATA REGISTER PORTD
DATA DIR. REG. PORTD
DATA REG. PORTG
DATA DIR. REG. PORTG
PORTE DRIVERS
PORTB DRIVERS
PORTD DRIVERS
PORTG DRIVERS
PE0 - PE7
PB0 - PB7
PD0 - PD7
PG0 - PG4

Timer ?? Counter ??

- Timer

➔ 사용자가 알고 있는 일정한 주파수의 시스템 clock을 입력으로 받아 이를 pre-scaler로 분주하고 counter로 계수 함으로서 정확한 시간 경과를 측정할 수 있는 동작.

- Counter

➔ CPU외부의 단자를 통하여 입력되는 미지의 clock 신호를 counter 로 계수하여 펄스 수 또는 주파수를 측정할 수 있는 동작.

ATmega128 has ....

8-bit Timer/Counter → 0,2
16-bit Timer/Counter → 1,3

8-Bit Timer/Counter 0

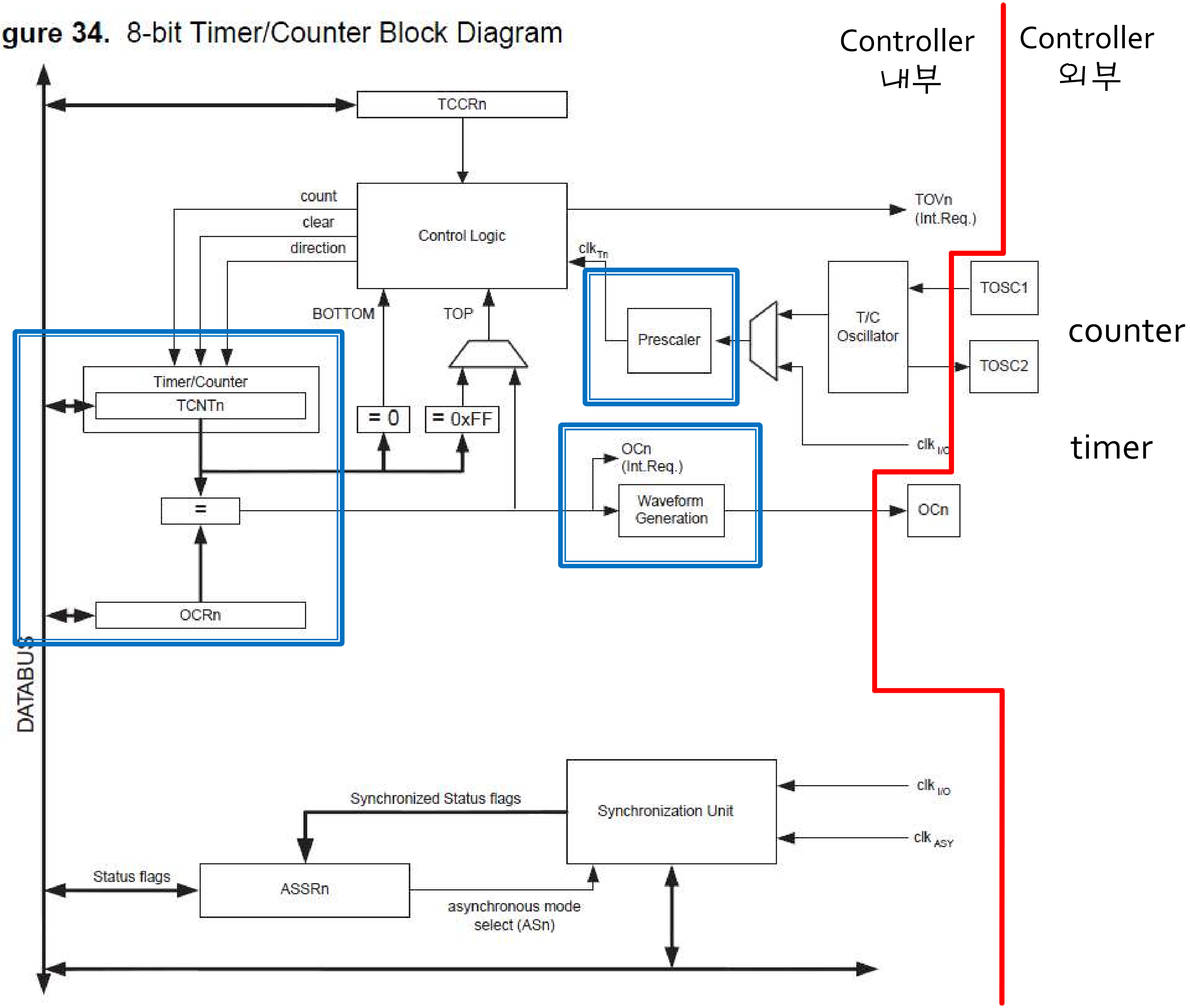


Figure 34. 8-bit Timer/Counter Block Diagram

Figure 45. Prescaler for Timer/Counter0

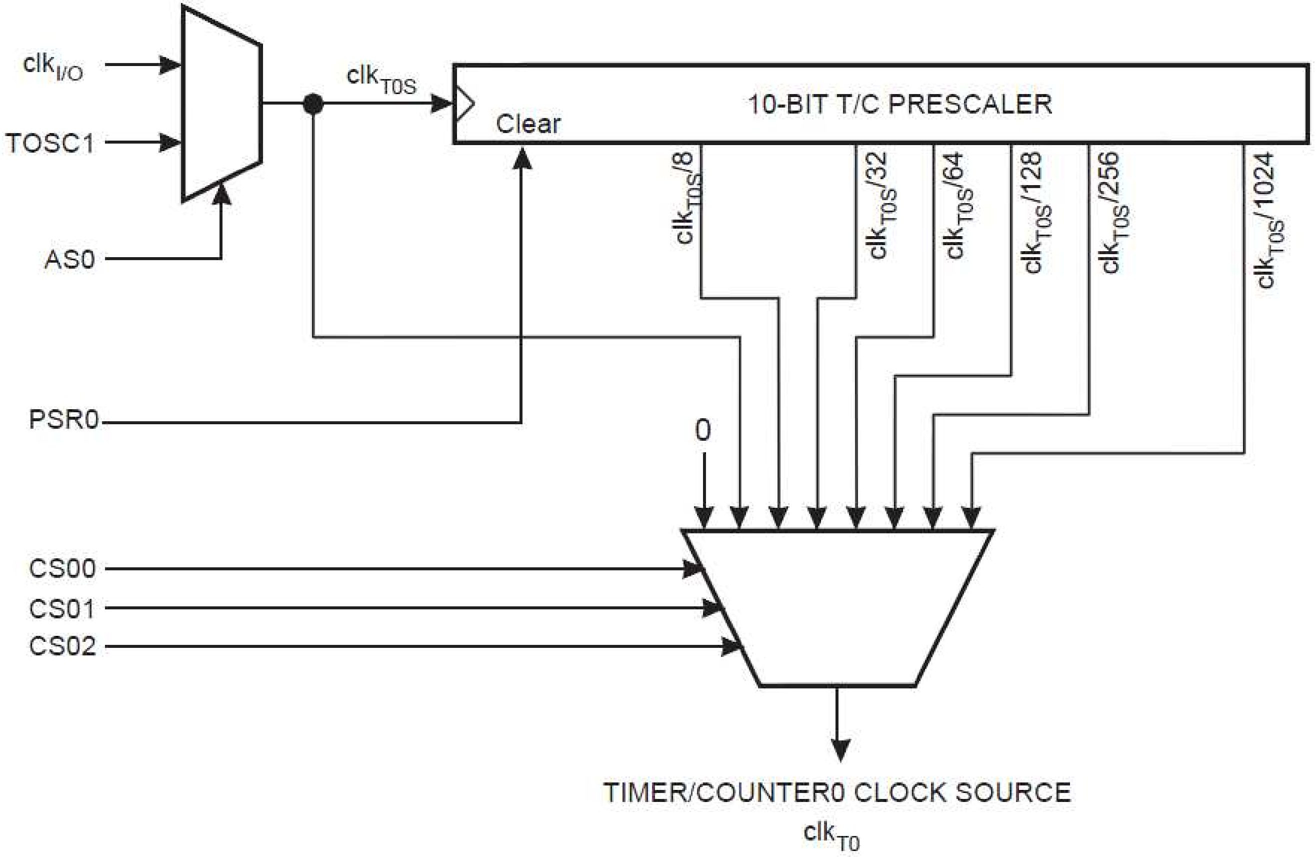


Figure 35. Counter Unit Block Diagram

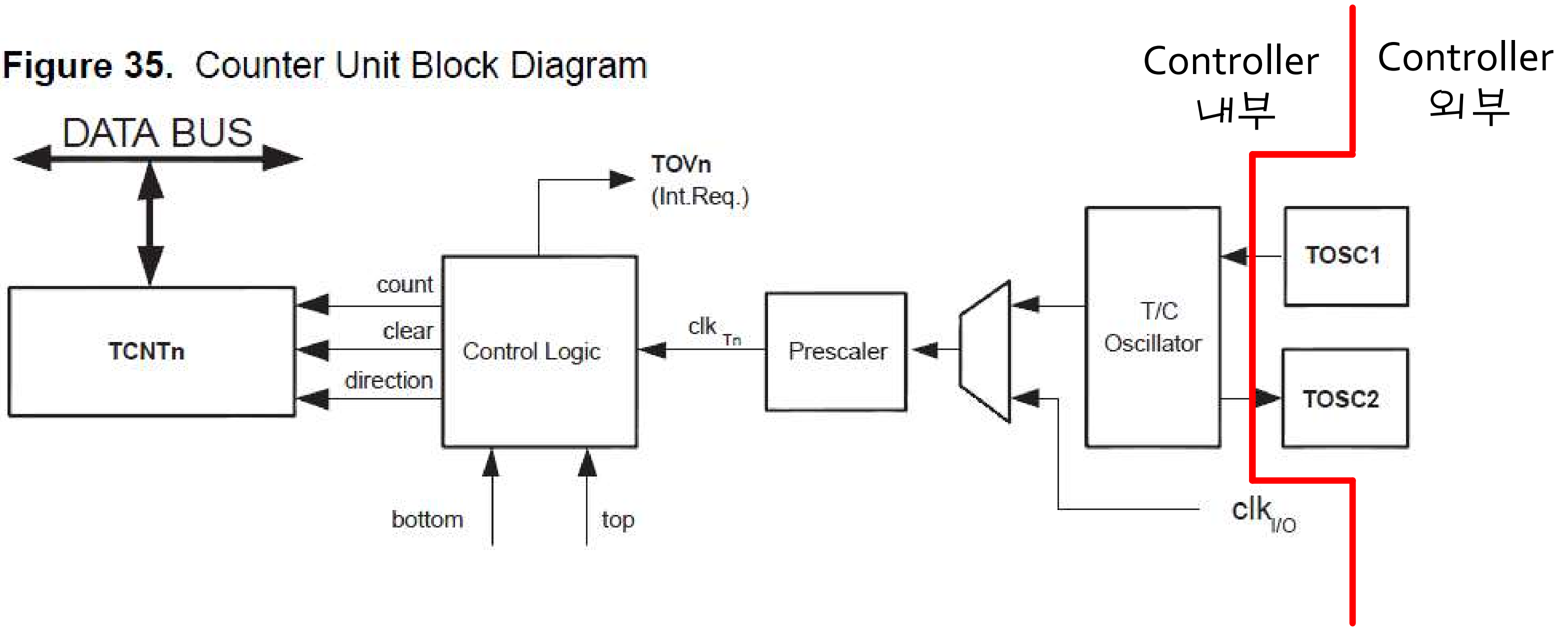
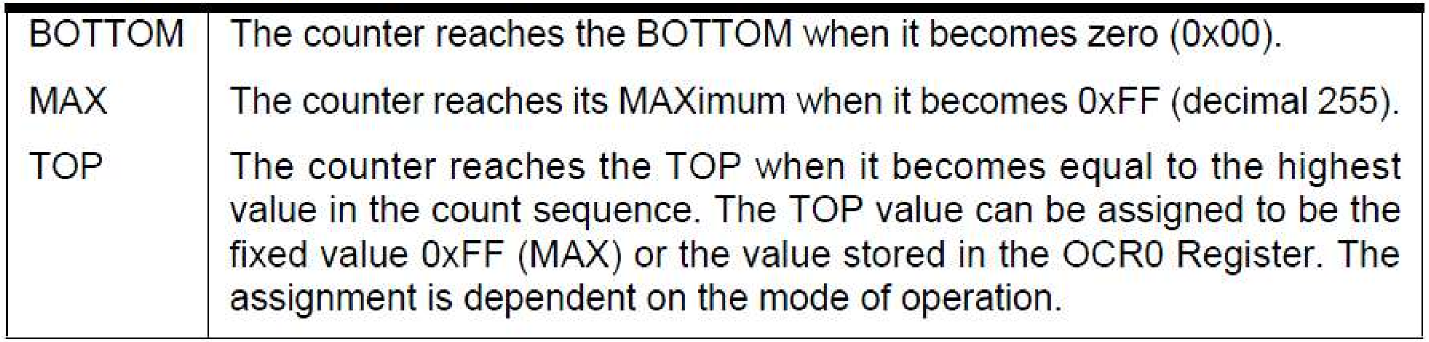


Table 51. Definitions

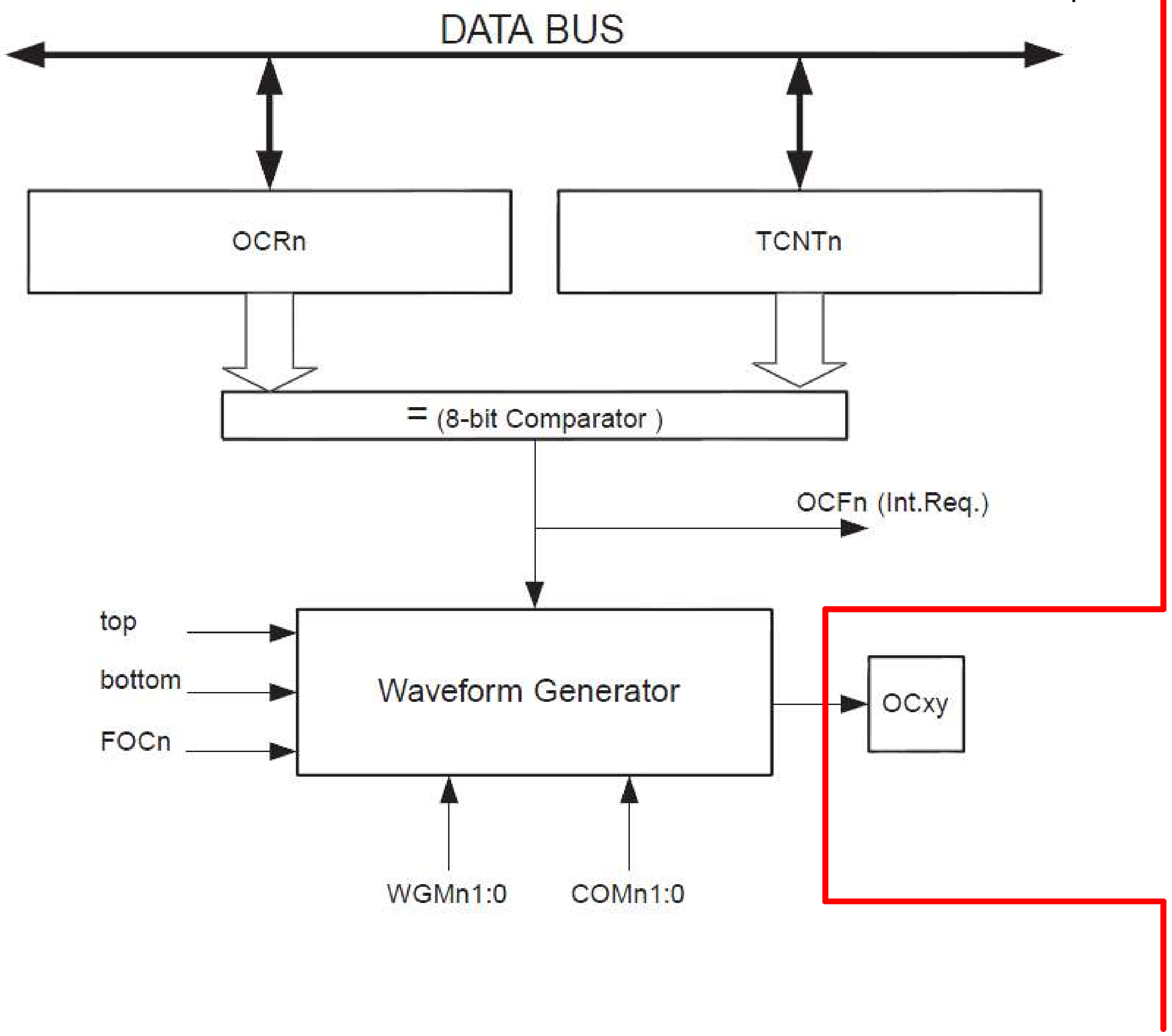


BOTTOM	The counter reaches the BOTTOM when it becomes zero (0x00).
MAX	The counter reaches its MAXimum when it becomes 0xFF (decimal 255).
TOP	The counter reaches the TOP when it becomes equal to the highest value in the count sequence. The TOP value can be assigned to be the fixed value 0xFF (MAX) or the value stored in the OCR0 Register. The assignment is dependent on the mode of operation.

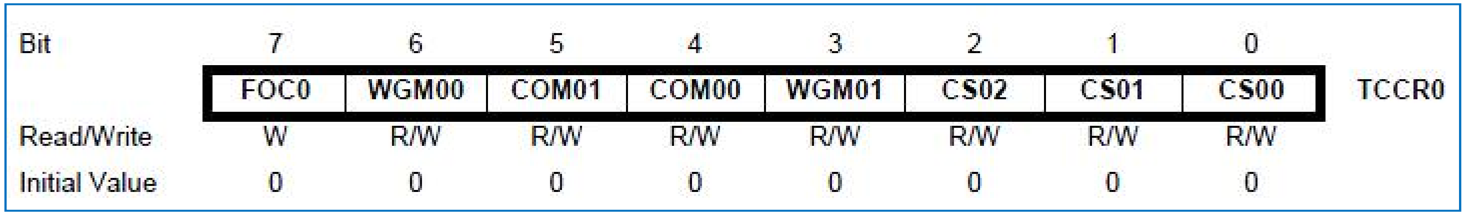
Figure 36. Output Compare Unit, Block Diagram

Controller 내부

Controller 외부

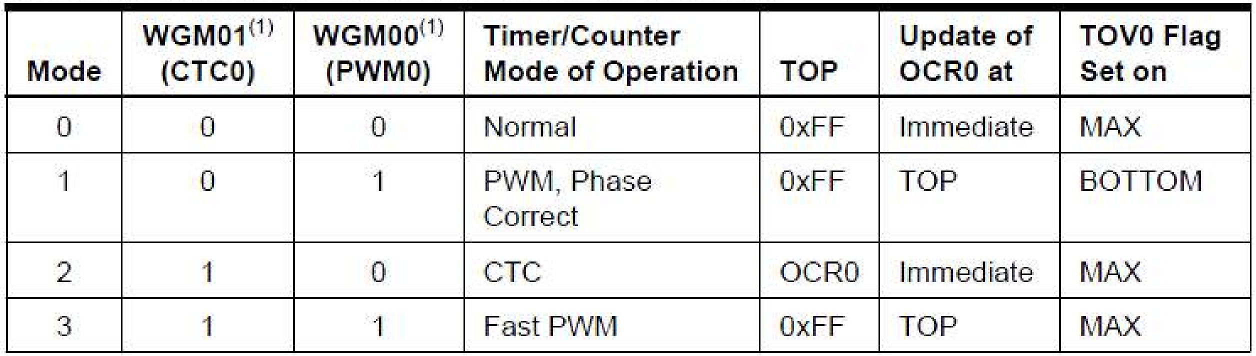


- Timer/Counter Control Register



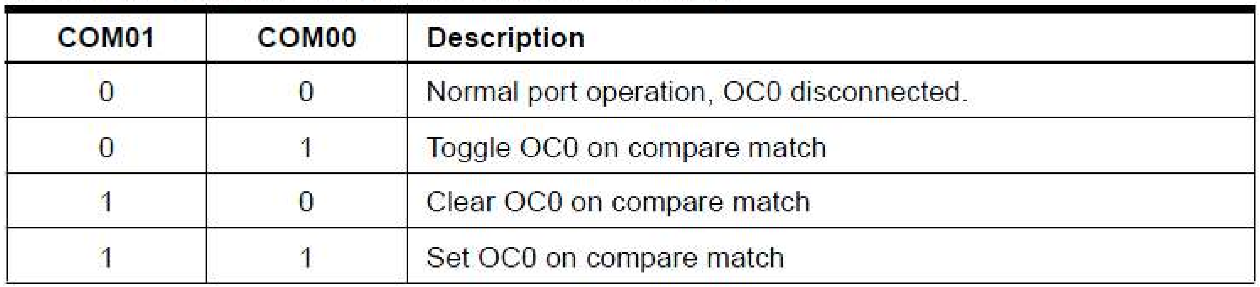
Bit	7	6	5	4	3	2	1	0	
	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Table 52. Waveform Generation Mode Bit Description



Mode	WGM01[1] (CTC0)	WGM00[1] (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0 at	TOV0 Flag Set on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	TOP	MAX

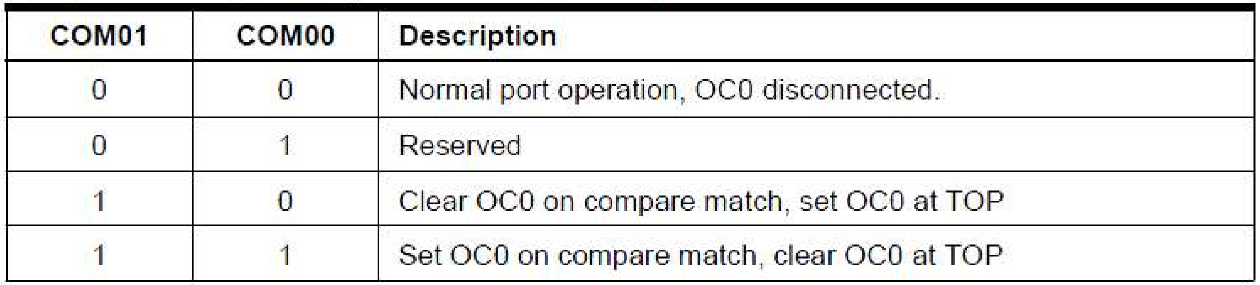
Table 53. Compare Output Mode, non-PWM Mode



COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Toggle OC0 on compare match
1	0	Clear OC0 on compare match
1	1	Set OC0 on compare match

WGM 설정 후 →

Table 54. Compare Output Mode, Fast PWM Mode[1]



COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match, set OC0 at TOP
1	1	Set OC0 on compare match, clear OC0 at TOP

WGM 설정 후 →

WGM 설정 후 →

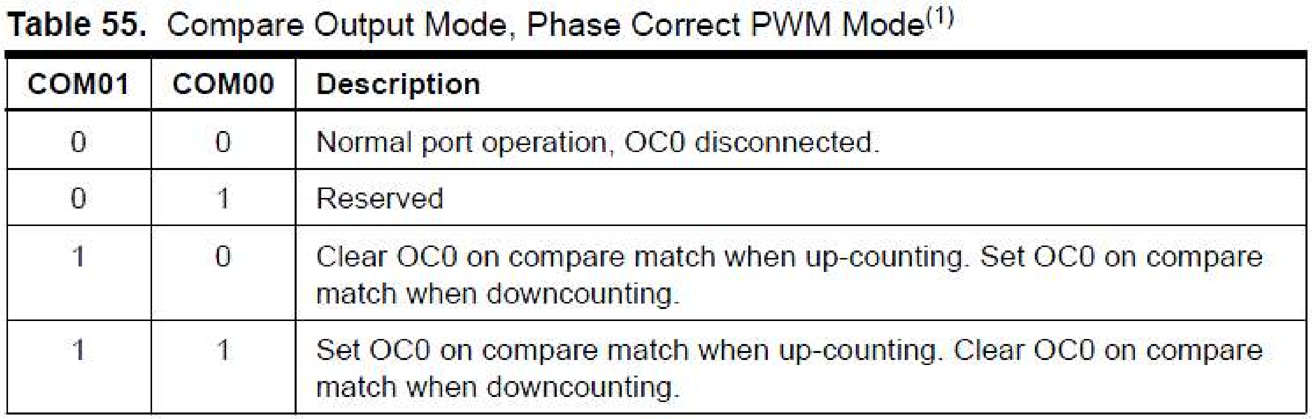


Table 55. Compare Output Mode, Phase Correct PWM Mode[1]

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match when up-counting. Set OC0 on compare match when downcounting.
1	1	Set OC0 on compare match when up-counting. Clear OC0 on compare match when downcounting.

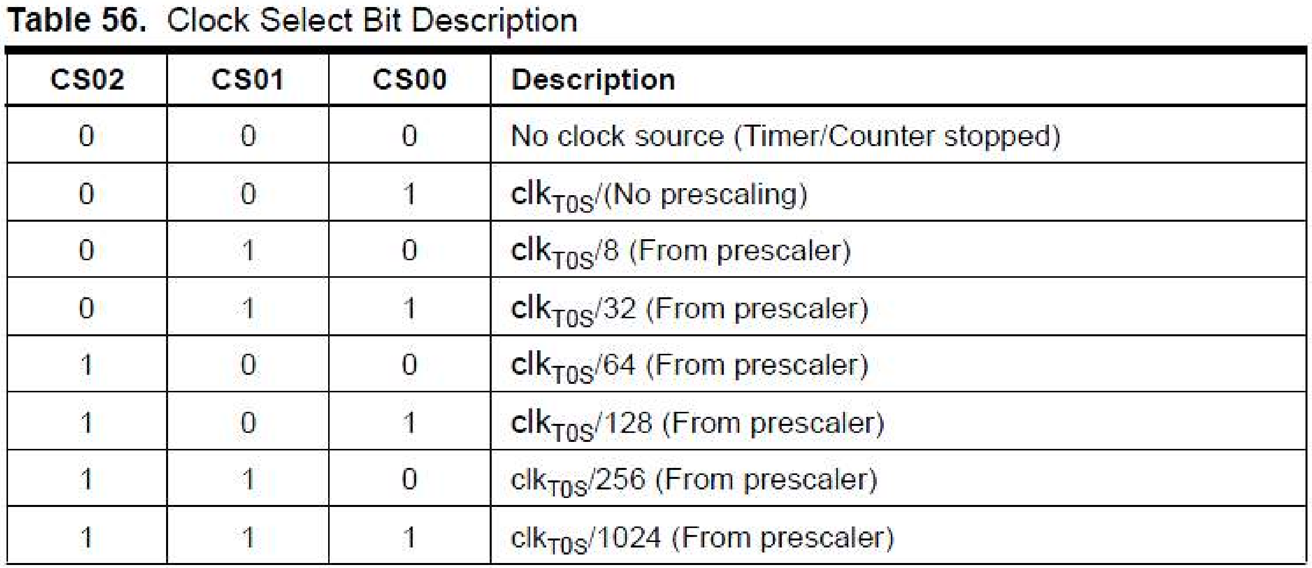
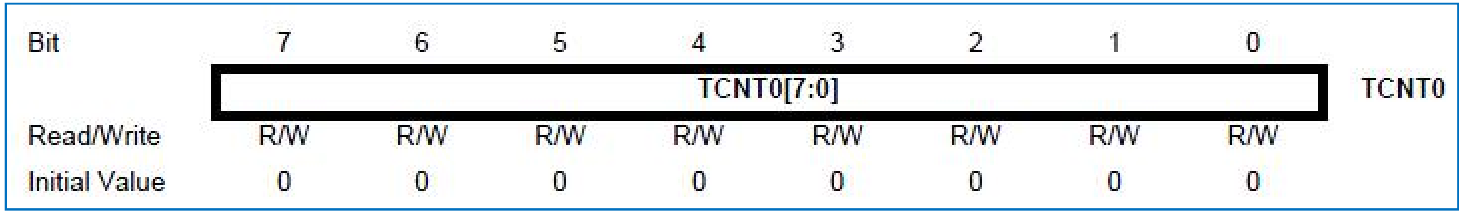


Table 56. Clock Select Bit Description

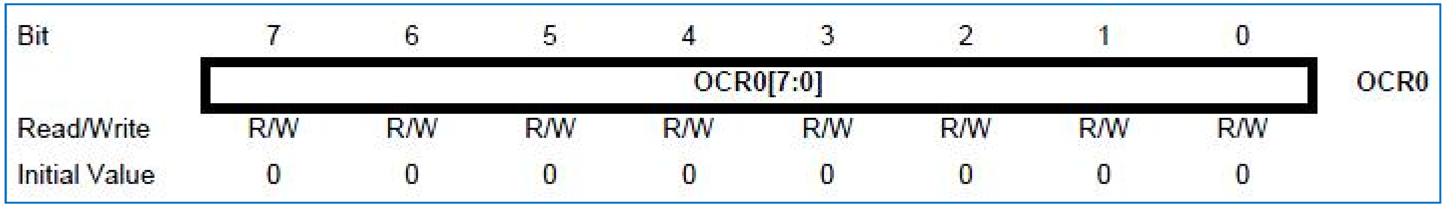
CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped)
0	0	1	clk_{T0S}/(No prescaling)
0	1	0	clk_{T0S}/8 (From prescaler)
0	1	1	clk_{T0S}/32 (From prescaler)
1	0	0	clk_{T0S}/64 (From prescaler)
1	0	1	clk_{T0S}/128 (From prescaler)
1	1	0	clk_{T0S}/256 (From prescaler)
1	1	1	clk_{T0S}/1024 (From prescaler)

- Timer/Counter Register



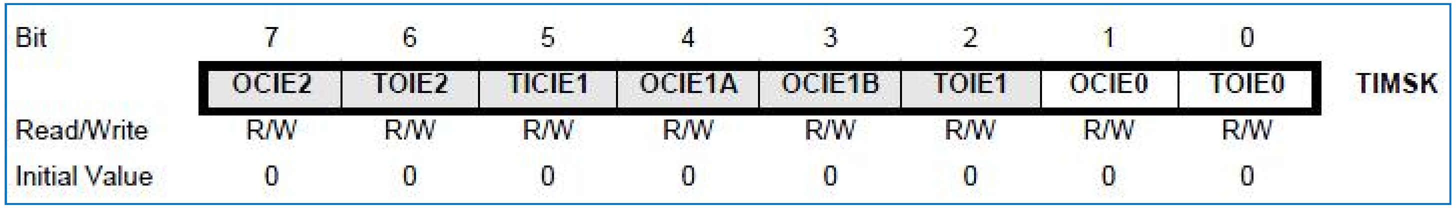
Bit	7	6	5	4	3	2	1	0	
	TCNT0[7:0]								TCNT0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Output Compare Register



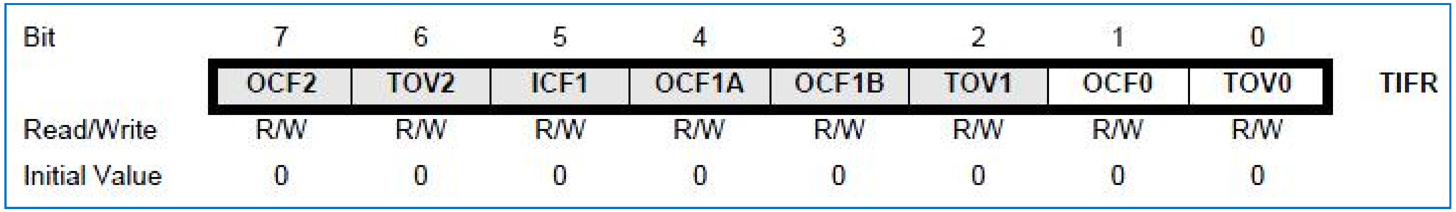
Bit	7	6	5	4	3	2	1	0	
	OCR0[7:0]								OCR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Timer/Counter Interrupt Mask Register



Bit	7	6	5	4	3	2	1	0	
	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	TIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Timer/Counter Interrupt Flag Register



Bit	7	6	5	4	3	2	1	0	
	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	TIFR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

16-Bit Timer/Counter 1

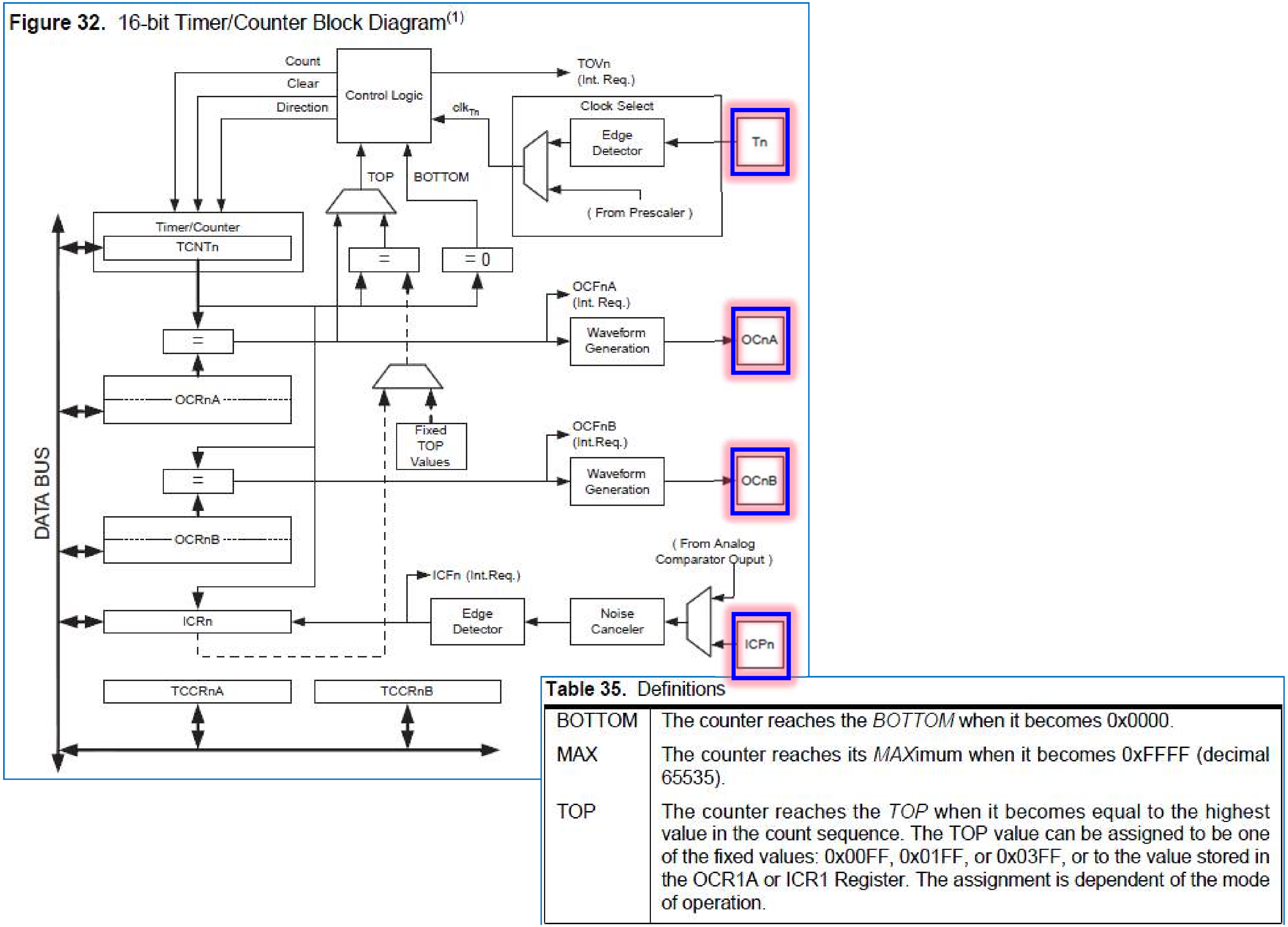


Figure 32. 16-bit Timer/Counter Block Diagram[1]

Table 35. Definitions

BOTTOM	The counter reaches the *BOTTOM* when it becomes 0x0000.
MAX	The counter reaches its *MAX*imum when it becomes 0xFFFF (decimal 65535).
TOP	The counter reaches the *TOP* when it becomes equal to the highest value in the count sequence. The TOP value can be assigned to be one of the fixed values: 0x00FF, 0x01FF, or 0x03FF, or to the value stored in the OCR1A or ICR1 Register. The assignment is dependent of the mode of operation.

Counter unit

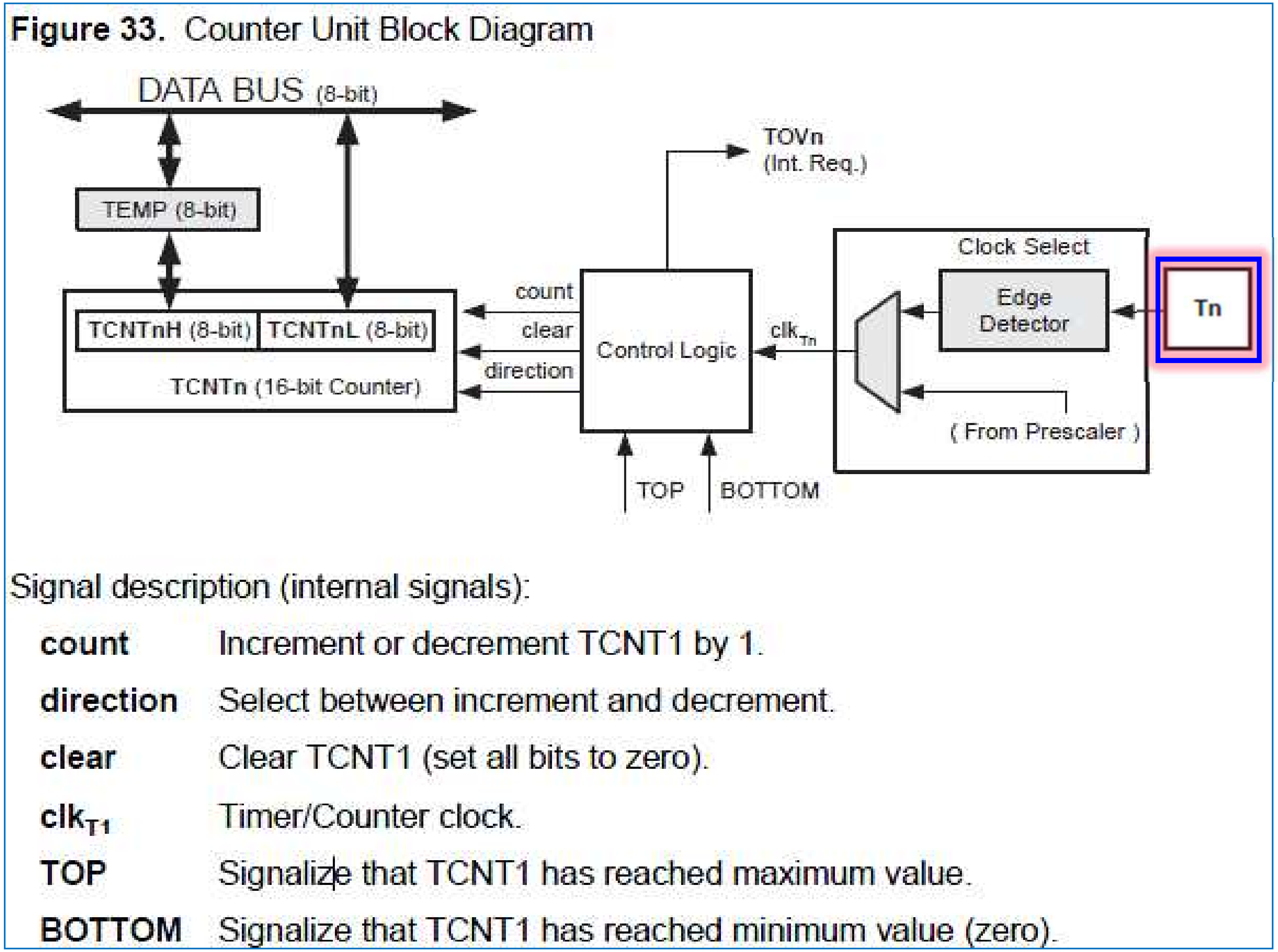


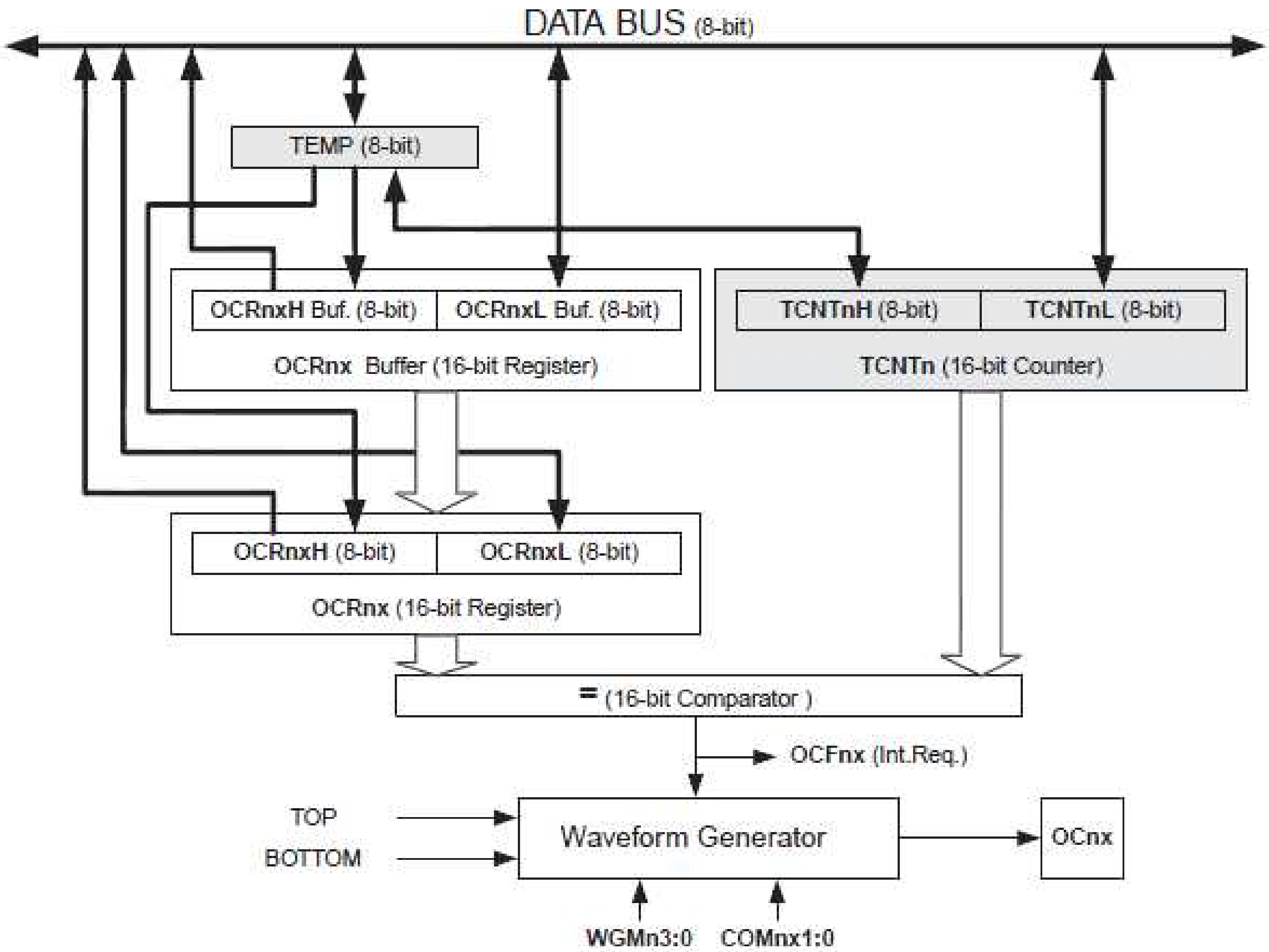
Figure 33. Counter Unit Block Diagram

Signal description (internal signals):

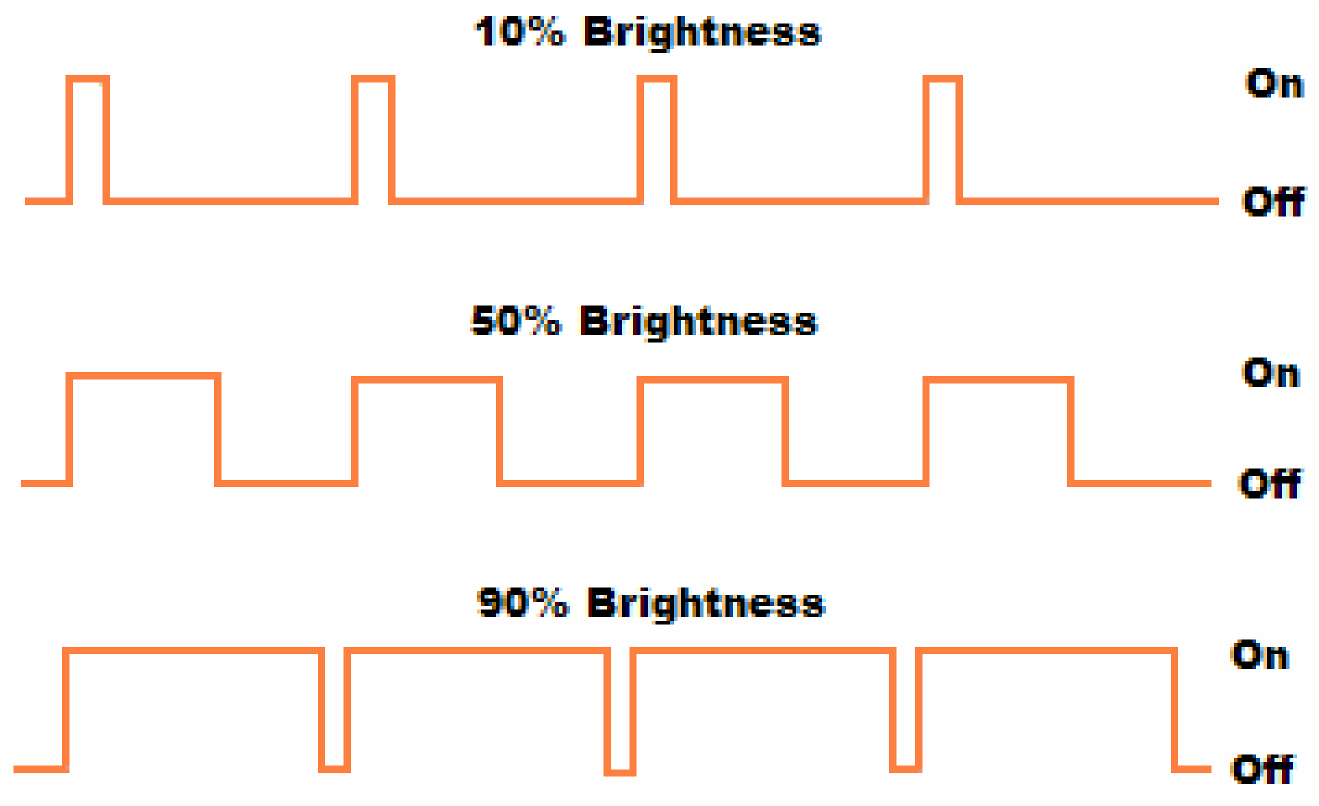
count	Increment or decrement TCNT1 by 1.
direction	Select between increment and decrement.
clear	Clear TCNT1 (set all bits to zero).
clk$_{T1}$	Timer/Counter clock.
TOP	Signalize that TCNT1 has reached maximum value.
BOTTOM	Signalize that TCNT1 has reached minimum value (zero).

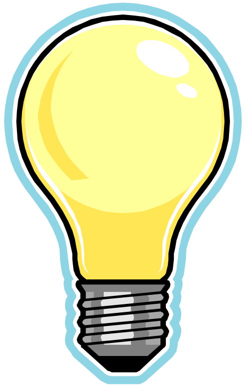
Output compare unit

Figure 35. Output Compare Unit, Block Diagram



PWM (Pulse Width Modulation)





- Period (= 1 / frequency)
- Width → Duty 계산(%)
- Level

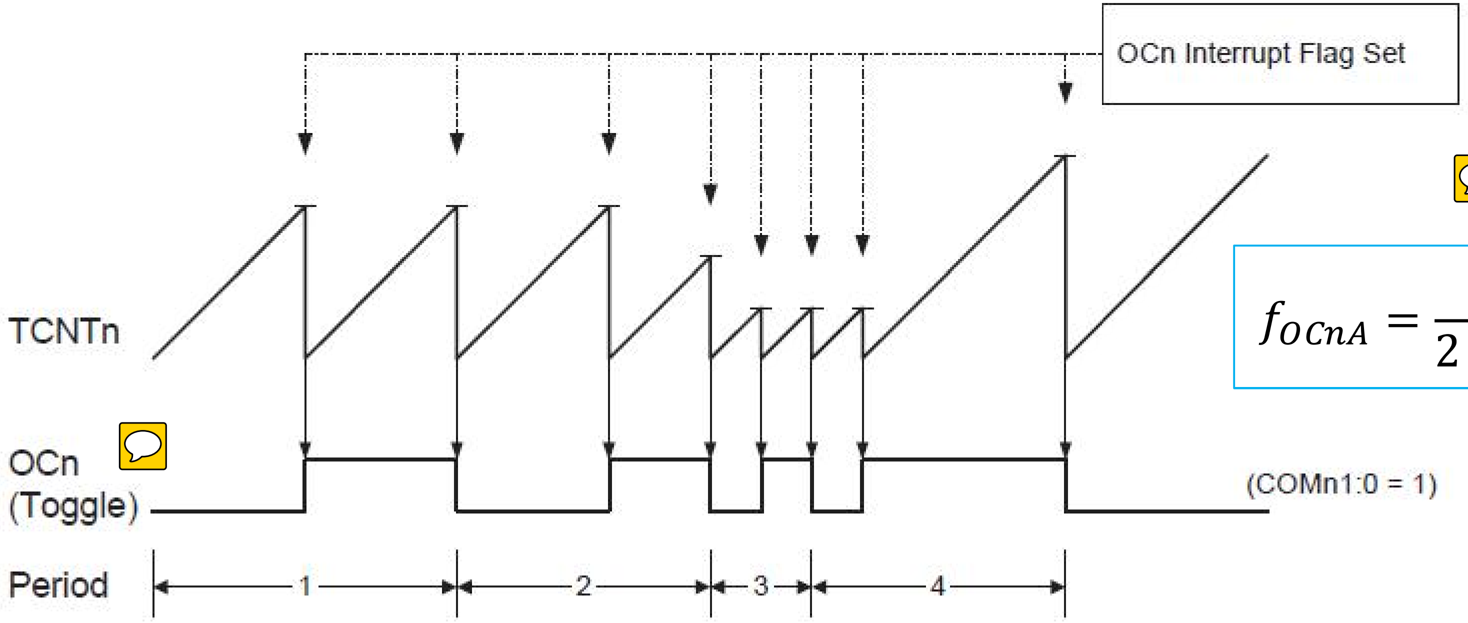
Mode of Operation

1. Normal Mode

- The simplest mode of operation is the *Normal* mode (WGM13:0 = 0).
- In this mode the counting direction is always up (incrementing)
- Maximum 16-bit value (MAX = 0xFFFF) and then restarts from the BOTTOM (0x0000).
- *Timer/Counter Overflow Flag* (TOV1) will be set in the same timer clock cycle as the TCNT1 becomes zero.

2. Clear Timer on Compare Match (CTC) Mode

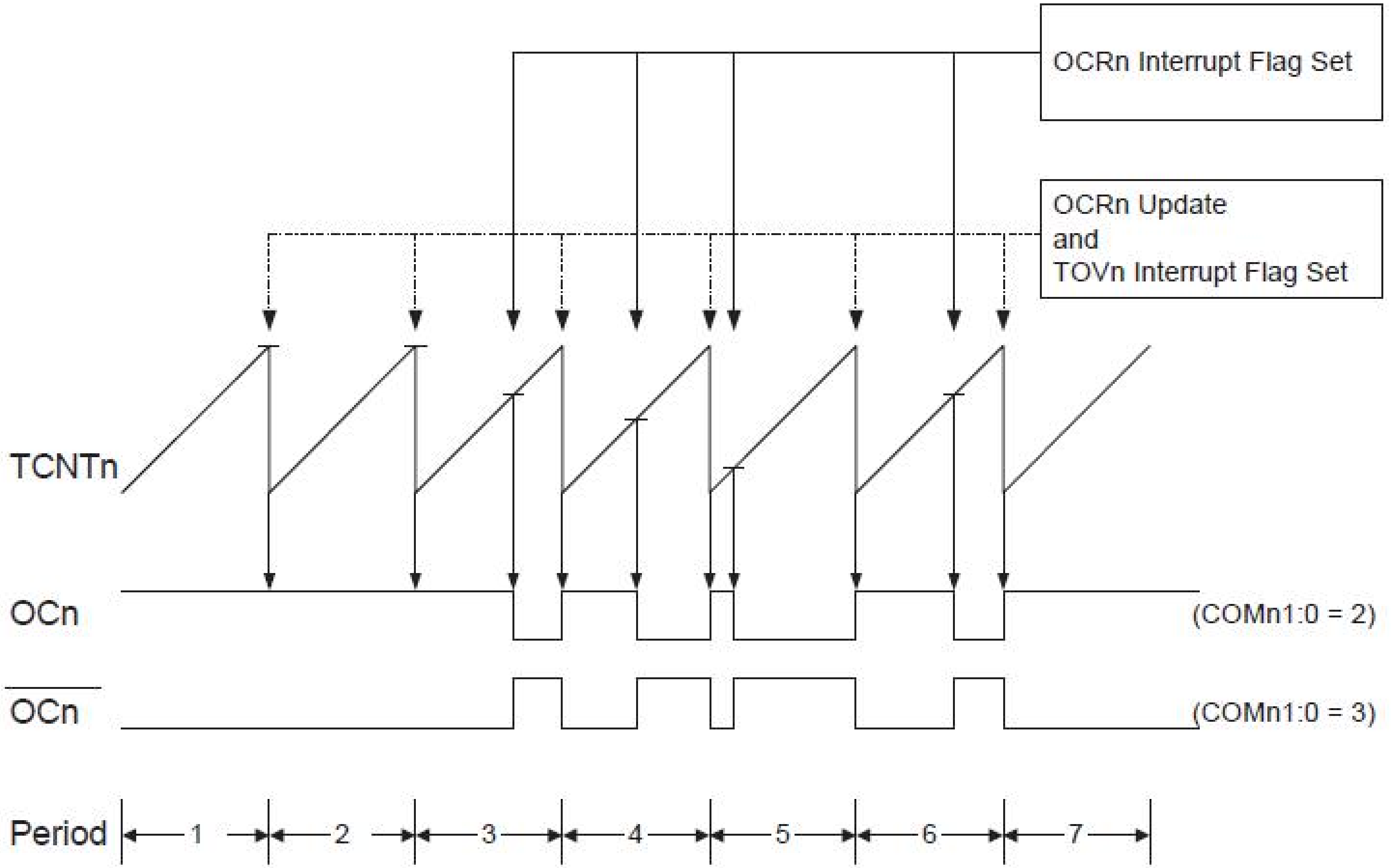
Figure 38. CTC Mode, Timing Diagram



$$f_{OCnA} = \frac{f_{clk_{I/O}}}{2 \cdot N \cdot (1 + OCRn)}$$

3. Fast PWM Mode

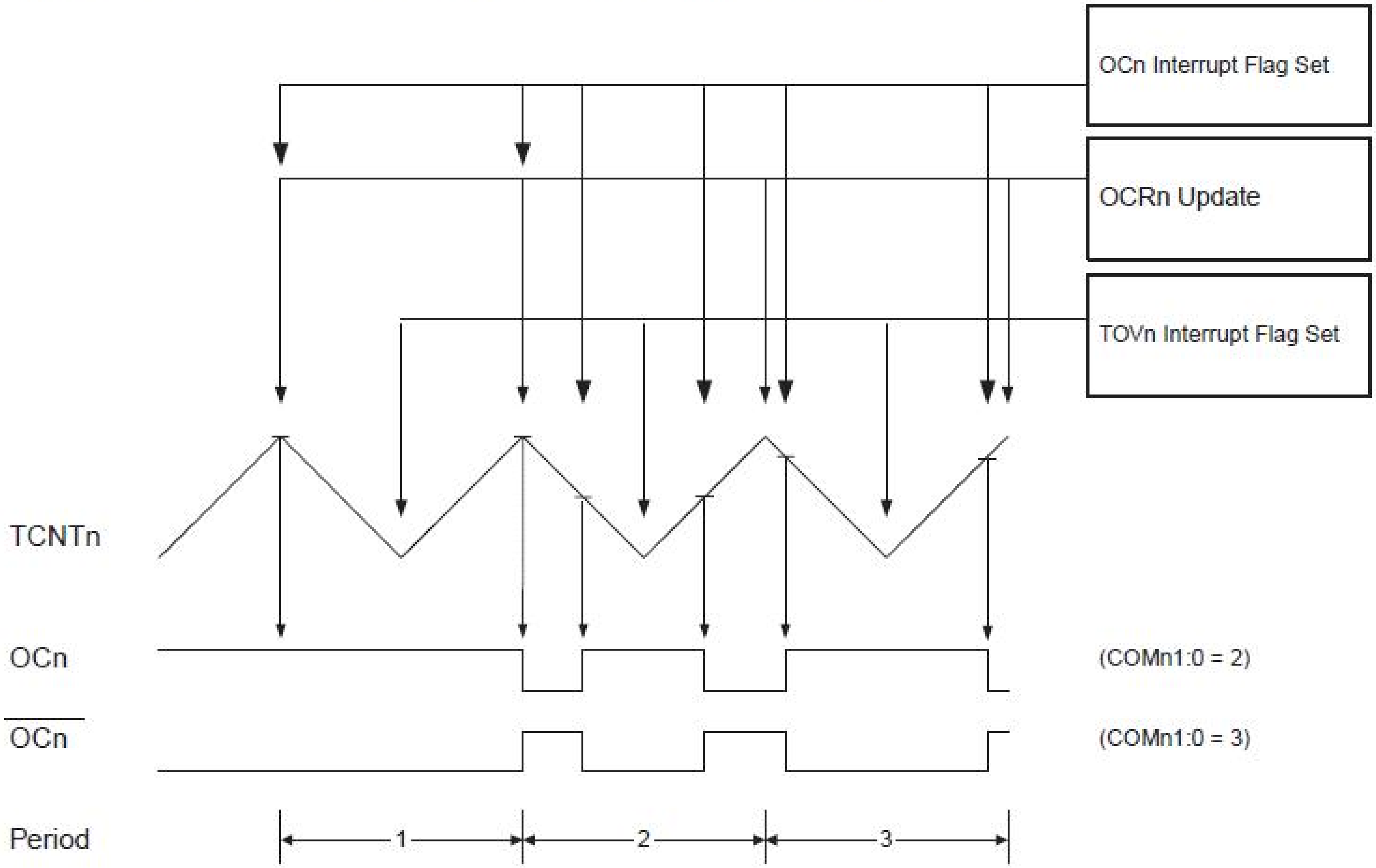
Figure 39. Fast PWM Mode, Timing Diagram



$$f_{OCnPW\ M} = \frac{f_{d\ k_{I/O}}}{N \cdot 256}$$

4. Phase Correct PWM Mode

Figure 40. Phase Correct PWM Mode, Timing Diagram



$$f_{OCnPCPWM} = \frac{f_{clk_{I/O}}}{N \cdot 510}$$

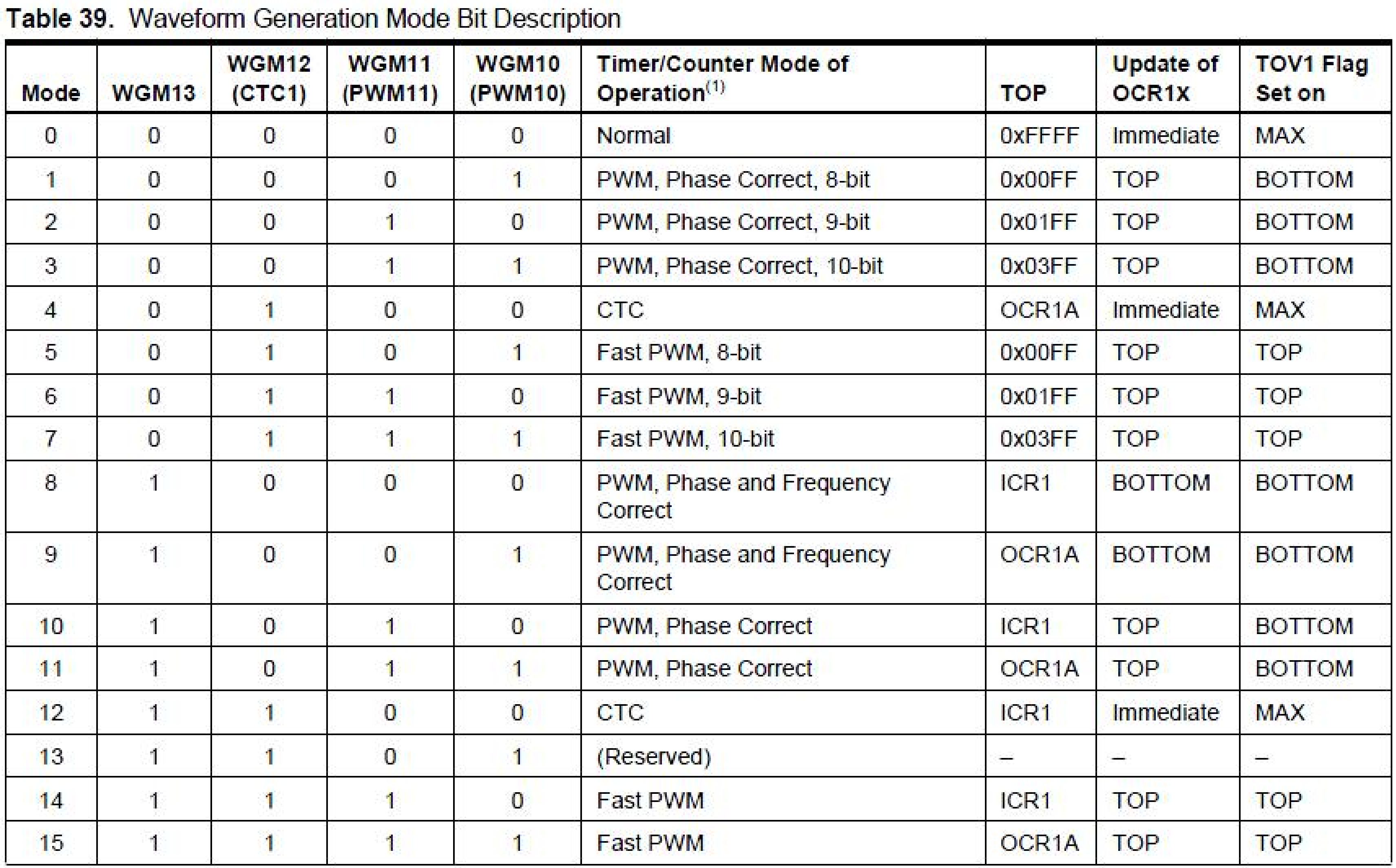
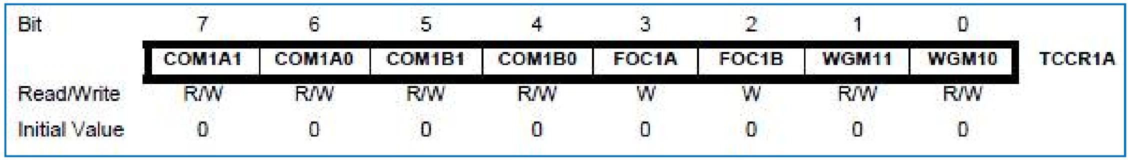


Table 39. Waveform Generation Mode Bit Description

Mode	WGM13	WGM12 (CTC1)	WGM11 (PWM11)	WGM10 (PWM10)	Timer/Counter Mode of Operation[(1)]	TOP	Update of OCR1X	TOV1 Flag Set on
0	0	0	0	0	Normal	0xFFFF	Immediate	MAX
1	0	0	0	1	PWM, Phase Correct, 8-bit	0x00FF	TOP	BOTTOM
2	0	0	1	0	PWM, Phase Correct, 9-bit	0x01FF	TOP	BOTTOM
3	0	0	1	1	PWM, Phase Correct, 10-bit	0x03FF	TOP	BOTTOM
4	0	1	0	0	CTC	OCR1A	Immediate	MAX
5	0	1	0	1	Fast PWM, 8-bit	0x00FF	TOP	TOP
6	0	1	1	0	Fast PWM, 9-bit	0x01FF	TOP	TOP
7	0	1	1	1	Fast PWM, 10-bit	0x03FF	TOP	TOP
8	1	0	0	0	PWM, Phase and Frequency Correct	ICR1	BOTTOM	BOTTOM
9	1	0	0	1	PWM, Phase and Frequency Correct	OCR1A	BOTTOM	BOTTOM
10	1	0	1	0	PWM, Phase Correct	ICR1	TOP	BOTTOM
11	1	0	1	1	PWM, Phase Correct	OCR1A	TOP	BOTTOM
12	1	1	0	0	CTC	ICR1	Immediate	MAX
13	1	1	0	1	(Reserved)	–	–	–
14	1	1	1	0	Fast PWM	ICR1	TOP	TOP
15	1	1	1	1	Fast PWM	OCR1A	TOP	TOP

This table is for Timer 1 and 3

- Timer/Counter 1 Control Register A



Bit	7	6	5	4	3	2	1	0	
	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	W	W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

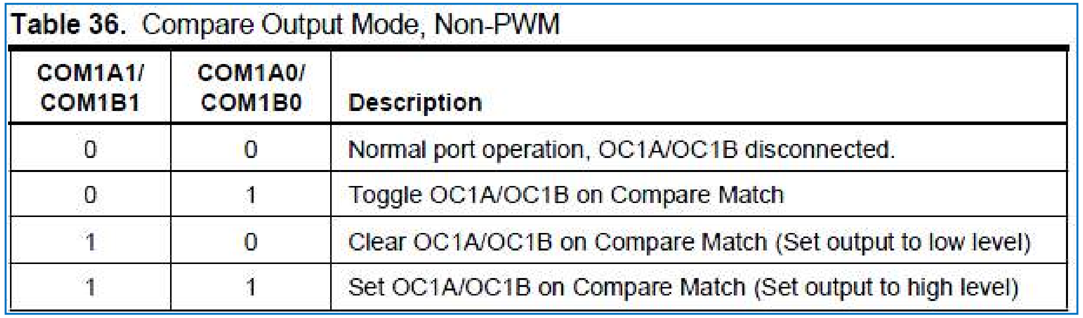


Table 36. Compare Output Mode, Non-PWM

COM1A1/ COM1B1	COM1A0/ COM1B0	Description
0	0	Normal port operation, OC1A/OC1B disconnected.
0	1	Toggle OC1A/OC1B on Compare Match
1	0	Clear OC1A/OC1B on Compare Match (Set output to low level)
1	1	Set OC1A/OC1B on Compare Match (Set output to high level)

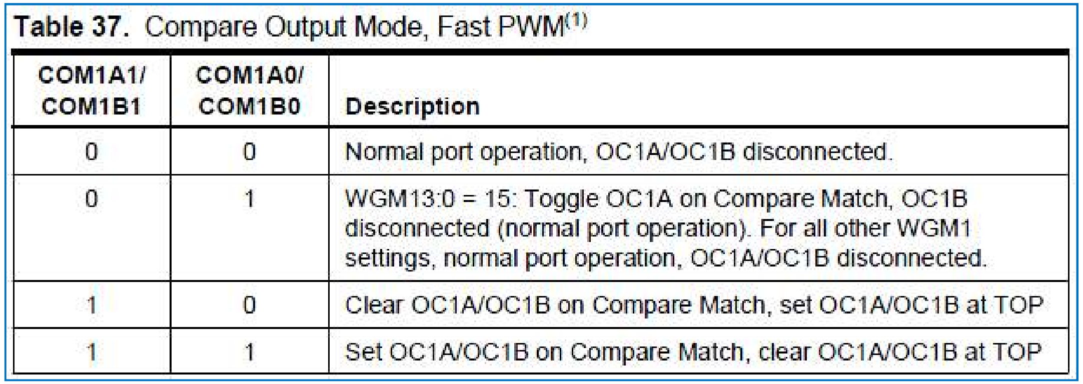


Table 37. Compare Output Mode, Fast PWM[(1)]

COM1A1/ COM1B1	COM1A0/ COM1B0	Description
0	0	Normal port operation, OC1A/OC1B disconnected.
0	1	WGM13:0 = 15: Toggle OC1A on Compare Match, OC1B disconnected (normal port operation). For all other WGM1 settings, normal port operation, OC1A/OC1B disconnected.
1	0	Clear OC1A/OC1B on Compare Match, set OC1A/OC1B at TOP
1	1	Set OC1A/OC1B on Compare Match, clear OC1A/OC1B at TOP

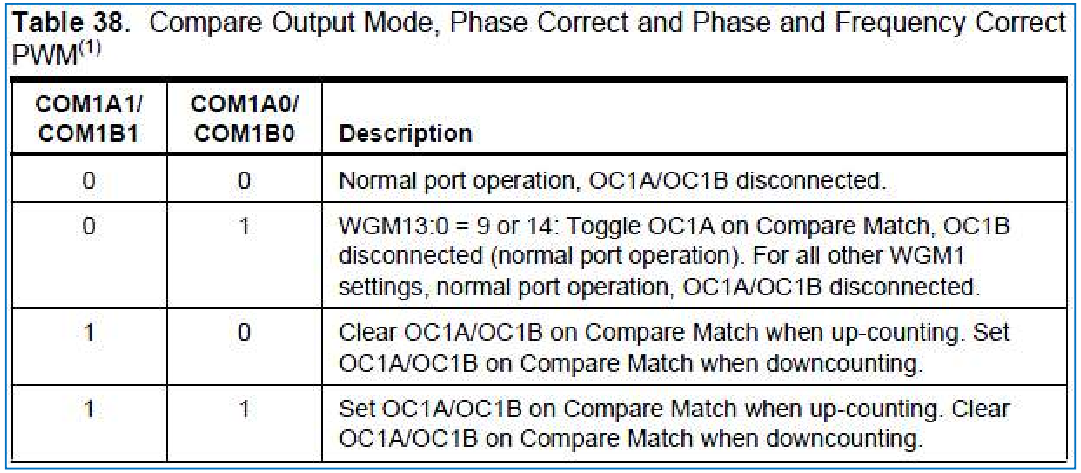
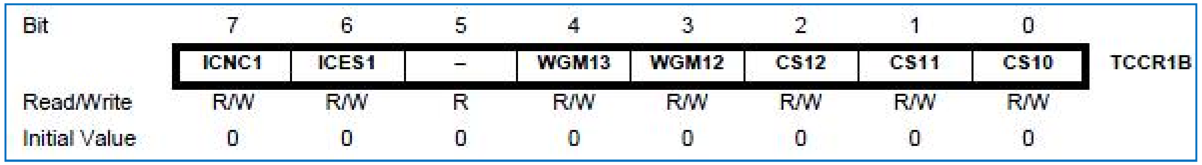


Table 38. Compare Output Mode, Phase Correct and Phase and Frequency Correct PWM[(1)]

COM1A1/ COM1B1	COM1A0/ COM1B0	Description
0	0	Normal port operation, OC1A/OC1B disconnected.
0	1	WGM13:0 = 9 or 14: Toggle OC1A on Compare Match, OC1B disconnected (normal port operation). For all other WGM1 settings, normal port operation, OC1A/OC1B disconnected.
1	0	Clear OC1A/OC1B on Compare Match when up-counting. Set OC1A/OC1B on Compare Match when downcounting.
1	1	Set OC1A/OC1B on Compare Match when up-counting. Clear OC1A/OC1B on Compare Match when downcounting.

- Timer/Counter 1 Control Register B



Bit	7	6	5	4	3	2	1	0	
	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

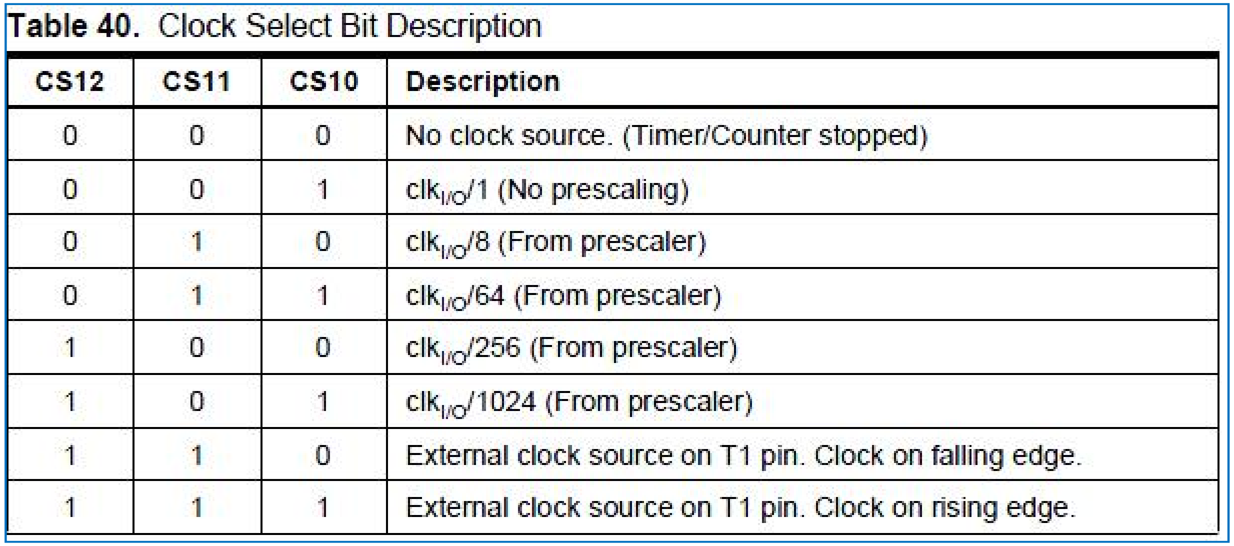
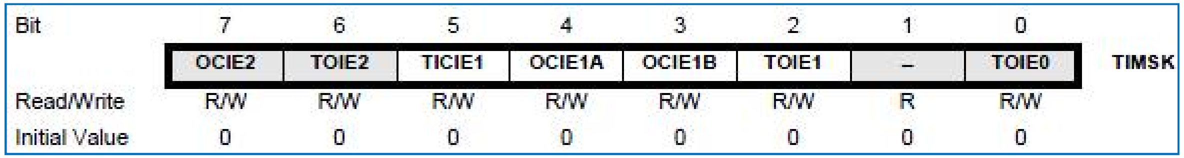


Table 40. Clock Select Bit Description

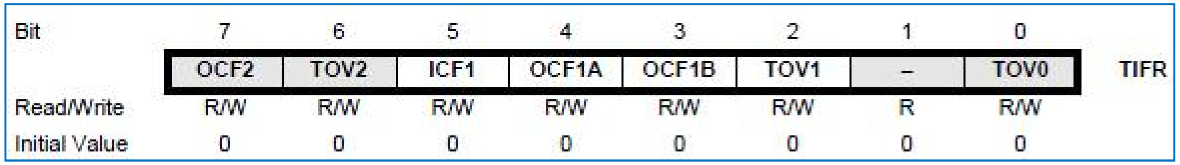
CS12	CS11	CS10	Description
0	0	0	No clock source. (Timer/Counter stopped)
0	0	1	$clk_{I/O}$/1 (No prescaling)
0	1	0	$clk_{I/O}$/8 (From prescaler)
0	1	1	$clk_{I/O}$/64 (From prescaler)
1	0	0	$clk_{I/O}$/256 (From prescaler)
1	0	1	$clk_{I/O}$/1024 (From prescaler)
1	1	0	External clock source on T1 pin. Clock on falling edge.
1	1	1	External clock source on T1 pin. Clock on rising edge.

- Timer/Counter Interrupt Mask Register



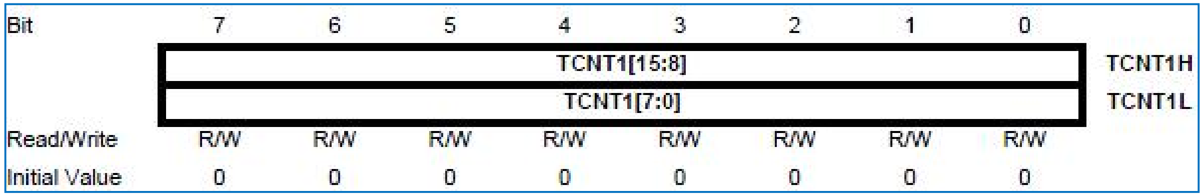
Bit	7	6	5	4	3	2	1	0	
	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	–	TOIE0	TIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Timer/Counter Interrupt Flag Register



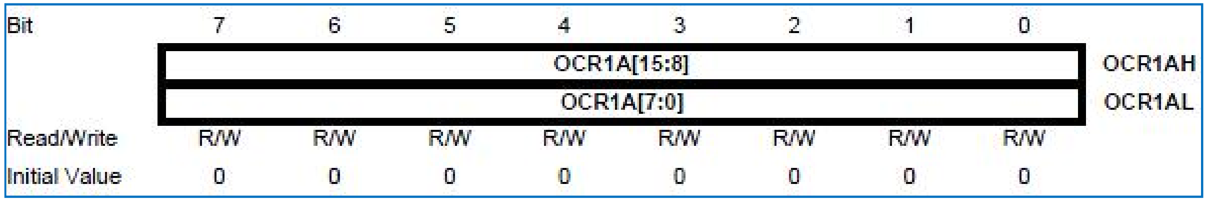
Bit	7	6	5	4	3	2	1	0	
	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	–	TOV0	TIFR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Timer/Counter 1



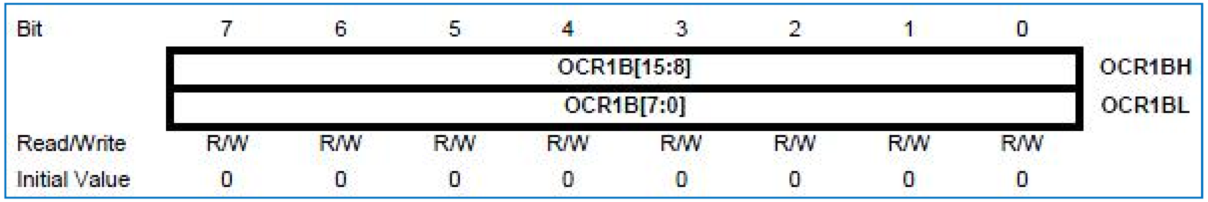
Bit	7	6	5	4	3	2	1	0	
	TCNT1[15:8]								TCNT1H
	TCNT1[7:0]								TCNT1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Output Compare Register 1 A



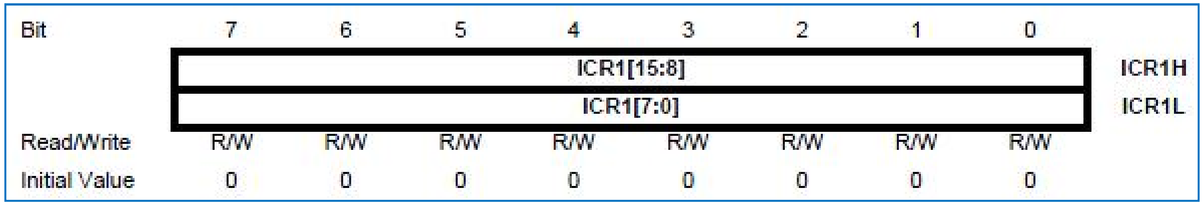
Bit	7	6	5	4	3	2	1	0	
	OCR1A[15:8]								OCR1AH
	OCR1A[7:0]								OCR1AL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Output Compare Register 1 B



Bit	7	6	5	4	3	2	1	0	
	OCR1B[15:8]								OCR1BH
	OCR1B[7:0]								OCR1BL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Input Capture Register 1



Bit	7	6	5	4	3	2	1	0	
	ICR1[15:8]								ICR1H
	ICR1[7:0]								ICR1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

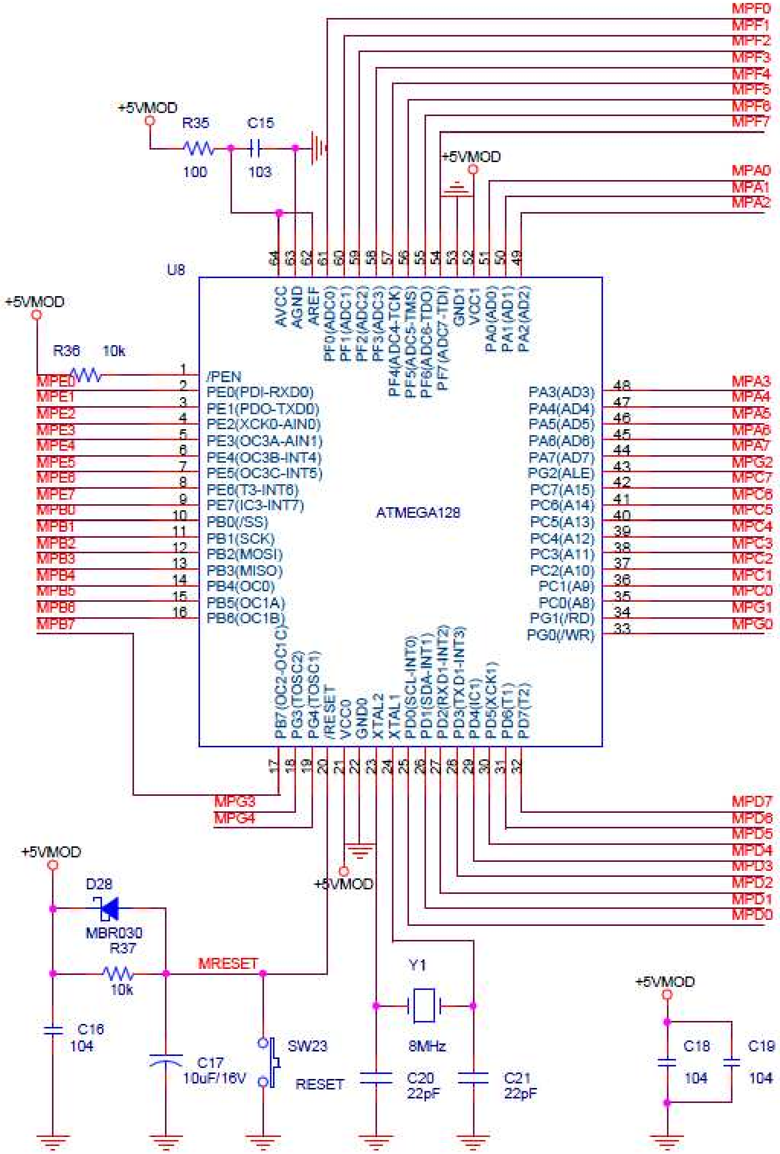
Chapter 6 - 예제 1

```
#include <iom128.h>
#include <intrinsics.h>

unsigned int data = 0xFE;

void main(void)
{
 DDRA = 0xFF;
 PORTA = 0xFF;

 TCCR0 = 0x07;
 TCNT0 = 0x00;
 TIMSK = 0x01;

 TIFR |= 0x01;

 __disable_interrupt();

 while(1)
 {
  while(!(TIFR&0x01));

  TIFR |= 0x01;
  TCNT0 = 0x00;

  PORTA = data;
  data--;
 }

}
```

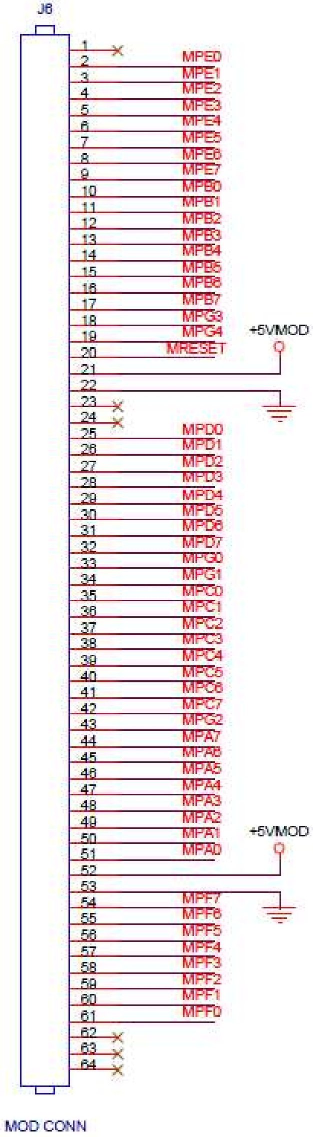
Chapter 6 - 예제 2

```
#include <iom128.h>
#include <intrinsics.h>

unsigned int data = 0xFE;

void main(void)
{
 DDRA = 0xFF;
 PORTA = 0xFF;

 TCCR0 = 0x07;
 TCNT0 = 0x00;
 TIMSK = 0x01;

 TIFR |= 0x01;

 __enable_interrupt();

 while(1)
 {
  PORTA = data;
 }
}

#pragma vector=TIMER0_OVF_vect
__interrupt void TIMER0_OVF_interrupt(void)
{
 __disable_interrupt();
 data--;
 __enable_interrupt();
}
```

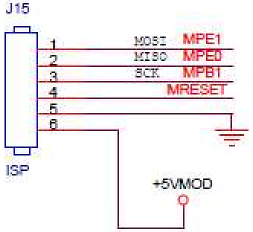
실험 Control Board



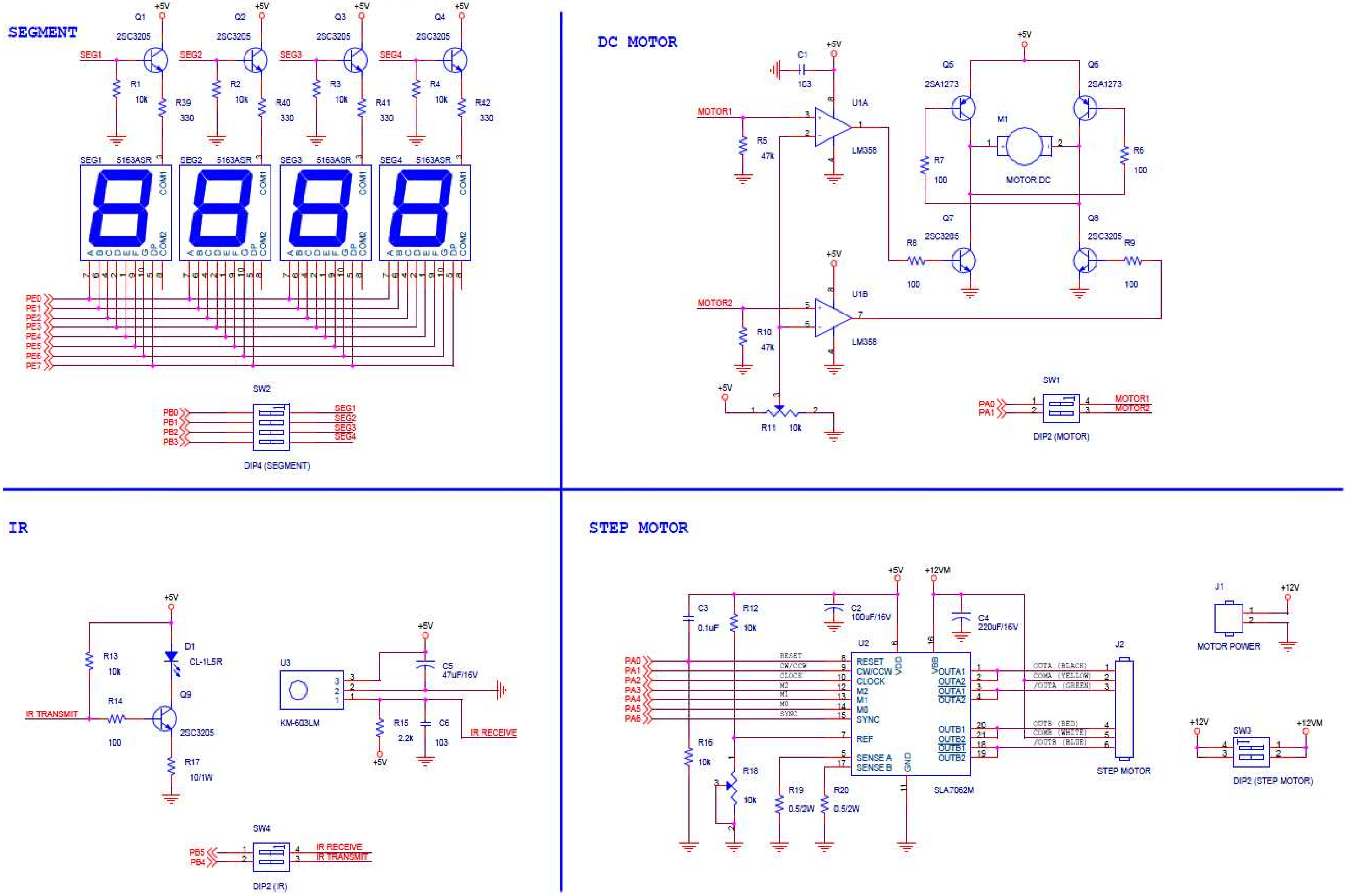
MAIN CONNECTOR MODULE



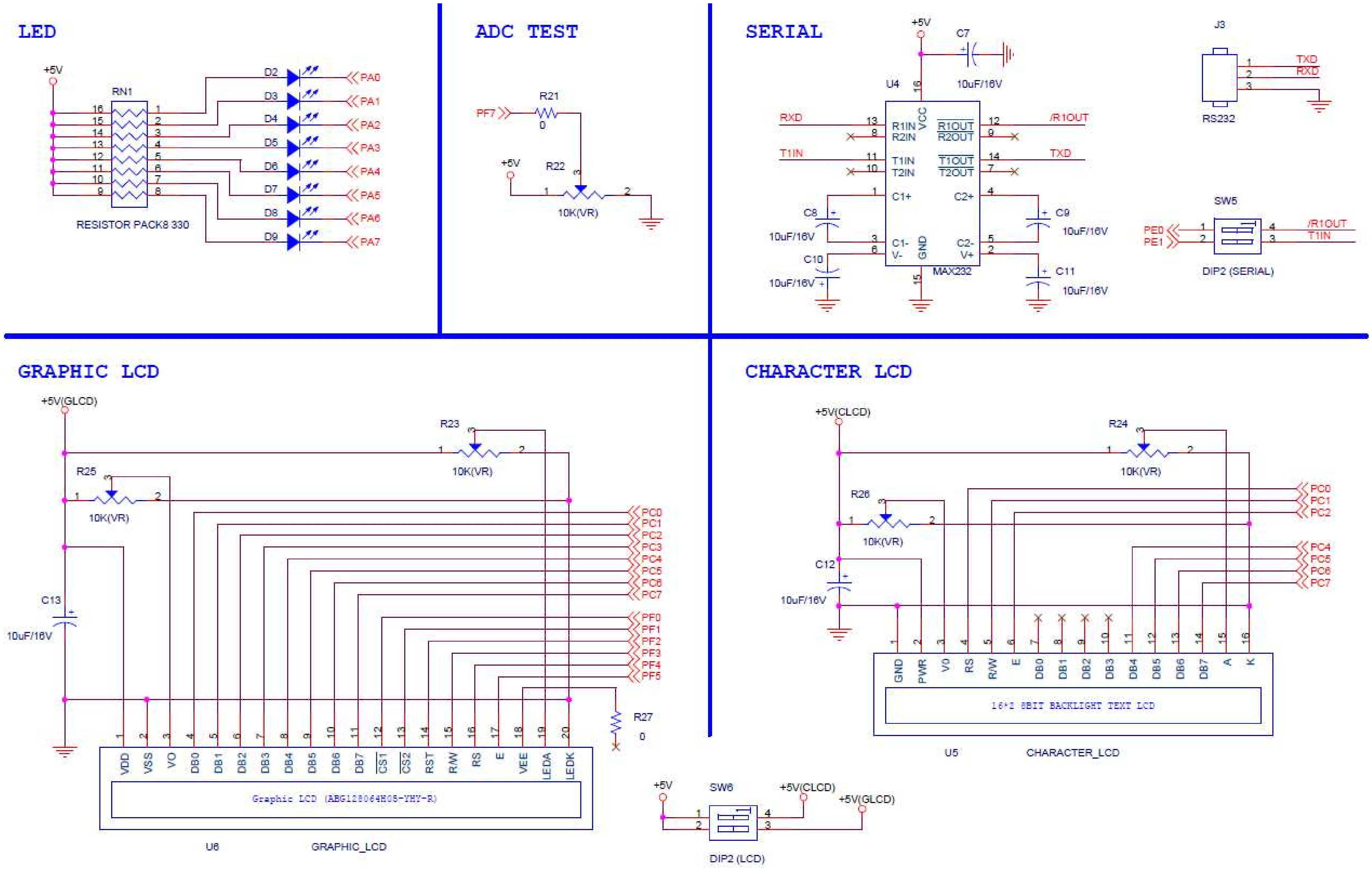
ISP



실험 Main Board 1

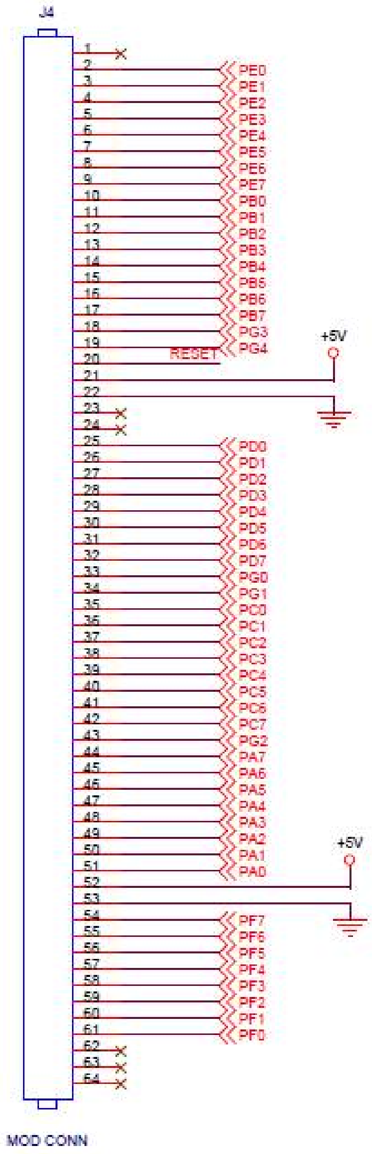


실험 Main Board 2

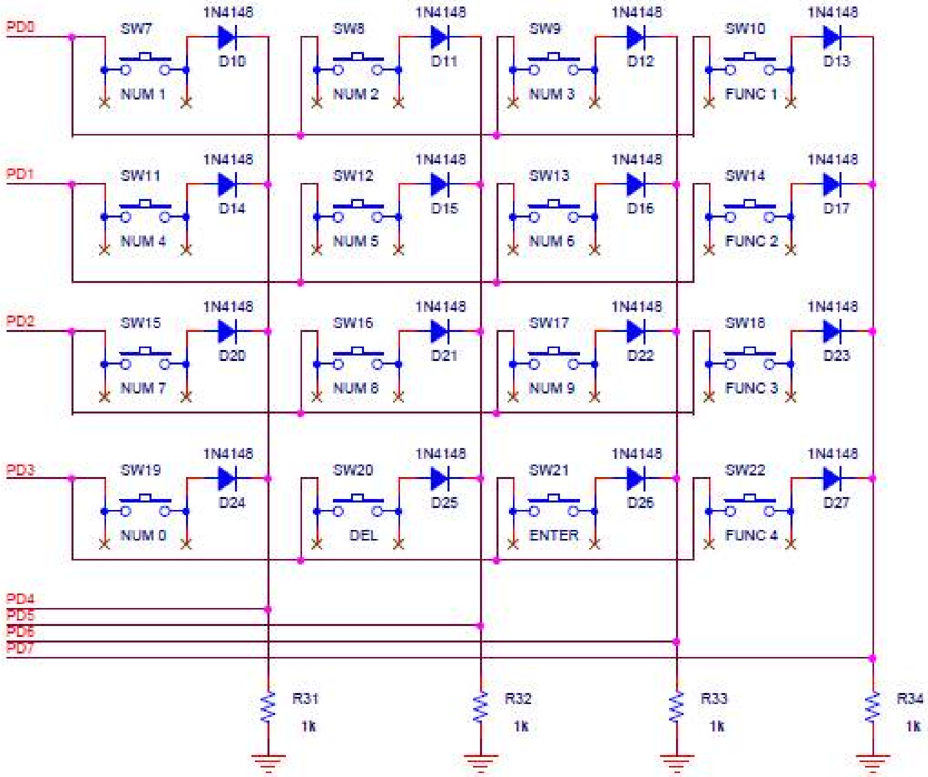


실험 Main Board 3

MAIN CONNECTOR BOARD



KEY-PAD



레포트

1. 실험 실습 완료 할 것(예제 6-1 ~ 6)

: 반드시 오실로스코프를 통하여 확인하고 이해할 것!!

: 예제 4,5,6 은 PB4와 PA0를 연결할 것

그리고, DDRA = 0xFE; 와 PORTA = 0xFE; 코드를 추가할 것