Electrical Engineering 372A
Dale Hetherington
Sample Exam 2b

1. The duty cycle of a pulse width modulation (PWM) signal is:
    a. The ratio of the high and low voltage signal
    b. The value of the high voltage of the PWM signal
    c. The ratio of the time that the signal is high to how long it takes to complete one cycle
    d. The ratio of the low voltage of the PWM signal.

2. A PWM signal should have a short period because:
    a. The faster a PWM signal pulses, the more likely it will be smoothed out by a capacitive load
    b. It can be either short or long period, it doesn't matter
    c. The period needs to be long
    d. To minimize interruptions of the microcontroller's CPU.

3. If you vary a signal between 0V and 5V and the effective voltage measured using a volt-meter is 4V, then the duty cycle is:
    a. 20%
    b. 60%
    c. 80%
    d. 40%

4. Timer 0 is configured for fast PWM mode with a prescaler of value of 1, and a system clock of 16 MHz. What is the frequency of the output?
    a. 1.0 kHz
    b. 7.8125 kHz
    c. 62.5 kHz
    d. 4 MHz

5. For Timer 0 fast PWM, Setting the COM0x1:0 bits to two will produce a non-inverted PWM and an inverted PWM output can be generated by setting the COM0x1:0 to three.

    a. TRUE

    b. FALSE

6. To generate a waveform frequency of 15,625 Hz using Timer 1 Fast PWM, 10-bit mode, what value of prescaler is needed? The system clock is 16 MHz.

    a. 1

    b. 8

    c. 64

    d. 256

    e. 1024

7. Timer 1 is configured to be in fast PWM 8-bit mode and use a non-inverting output. To set the duty cycle to 25% the value of OCR1A should be set to

    a. 31

    b. 63

    c. 127

    d. 255

8. According to the Arduino Mega2560 schematic, the pin on the development board associated with output C for output compare module 4 is

    a. 3

    b. 8

    c. 6

    d. 2

9. A program you inherited uses Fast PWM mode with the value in OCR1A set to the TOP value. You are checking to ensure that the register bits for WGM13, WGM12, WGM11, and WGM10 are set correctly. They should read:

    a. WGM13 = 0, WGM12 = 1, WGM11 = 0, WGM10 = 1
    b. WGM13 = 1, WGM12 = 1, WGM11 = 1, WGM10 = 1
    c. WGM13 = 0, WGM12 = 1, WGM11 = 0, WGM10 = 0
    d. WGM13 = 0, WGM12 = 1, WGM11 = 1, WGM10 = 0

10. When generating a PWM waveform from an output compare, the corresponding Data Direction register should be set to input during the initialization sequence.

    a. TRUE
    b. FALSE

11. In generating a PWM waveform, the TOP value determines the

    a. Duty cycle
    b. Frequency
    c. Prescaler
    d. Interrupt signal

12. An ADC that can digitize 256 different levels has what bit-depth?

    a. 7
    b. 6
    c. 5
    d. 8

13. An ADC with a bit-depth of 12 can represent how different levels?

    a. 4096
    b. 2048
    c. 512
    d. 256

14. If we have an 8-bit ADC, V_high = 2V, V_low = 0 V, and we receive a digital value of 10, what voltage does this correspond to?

    a. 312 mV

    b. 156 mV

    c. 78 mV

    d. 7 mV

15. An analog input signal has a maximum frequency component of 40 kHz. What is the minimum sampling frequency that the ADC should be set at so that all bandwidth information is digitized?

    a. 100 kHz

    b. 80kHz

    c. 40 kHz

    d. 20 kHz

16. When reading the result from the ADC, register ADCH should be read before ADCL.

    a. TRUE

    b. FALSE

17. Which line of code will configure the ADC to immediately begin the conversion process and continue doing so as fast as it can, assuming the ADTS bits are configured to put the ADC into "Free Running" mode?

    a. ADCSRA |= (1 << ADEN | 1 << ADSC);

    b. ADCSRA |= (1 << ADEN | 1 << ADATE);

    c. ADCSRA |= (1 << ADEN);

    d. ADCSRA |= (1 << ADEN | 1 << ADSC | 1 << ADATE);

18. How long does one ADC conversion take for an ADC clock cycle of 200 kHz?

    a. 1 ms
    b. 300 μs
    c. 65 μs
    d. 52 μs

19. An asynchronous protocol is defined by:

    a. A clock signal is sent from transmitter to receiver
    b. A clock signal is not sent from transmitter to receiver

20. Given that the first byte sent from the master is 0b11011010, what is the address of the I2C device this transmission is meant for?

    a. 0b1101000
    b. 0b1101101
    c. 0b1011010
    d. 0b0101110

21. A half-duplex protocol is one where a device can transmit and receive simultaneously

    a. TRUE
    b. FALSE

22. In I2C, the START and STOP conditions are signalled by changing the level of the SDA line when the SCL line is high.

    a. TRUE
    b. FALSE

23. Given the internal clock speed is 16 MHz, and the USART module is initialized to normal mode, what is the value of UBRR0 will create a baud rate of 9600?

    a. UBRR0 = 103
    b. UBRR0 = 128
    c. UBRR0 = 1024
    d. UBRR0 = 64

24. If the USART if configured to work in an 8-E-1 configuration and the data that is sent in binary is 0b10110011, what should the value of the parity bit be:

    a. 1
    b. 0
    c. There is no parity bit

25. After the last byte has been received by the Master in I2C, the Master should

    a. take possession of the SDA line after the last received byte
    b. keep waiting for another byte to be transferred
    c. transition into sleep mode to conserve power
    d. inform the Slave Transmitter by sending a Not Acknowledge (NACK) after the last received data byte

PWM formulas

$$V_{eff} = \frac{(V_{high} - V_{low})T_{pulse}}{T_{period}} + V_{low}$$

$$Duty\ cycle = \frac{T_{pulse}}{T_{period}} x\ 100\ (\%)$$

Timer 0

Table 16-8. Waveform Generation Mode Bit Description

Mode	WGM2	WGM1	WGM0	Timer/Counter Mode of Operation	TOP	Update of OCRx at	TOV Flag Set on(1)(2)
0	0	0	0	Normal	0xFF	Immediate	MAX
1	0	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	0	1	0	CTC	OCRA	Immediate	MAX
3	0	1	1	Fast PWM	0xFF	TOP	MAX
4	1	0	0	Reserved	–	–	–
5	1	0	1	PWM, Phase Correct	OCRA	TOP	BOTTOM
6	1	1	0	Reserved	–	–	–
7	1	1	1	Fast PWM	OCRA	BOTTOM	TOP

Note: 1. MAX = 0xFF
2. BOTTOM = 0x00

Timer 0 CTC Mode

For generating a waveform output in CTC mode, the OC0A output can be set to toggle its logical level on each Compare Match by setting the Compare Output mode bits to toggle mode (COM0A1:0 = 1). The OC0A value will not be visible on the port pin unless the data direction for the pin is set to output. The waveform generated will have a maximum frequency of $f_{OC0} = f_{clk\_I/O}/2$ when OCR0A is set to zero (0x00). The waveform frequency is defined by the following equation:

$$f_{OCnx} = \frac{f_{clk\_I/O}}{2 \cdot N \cdot (1 + OCRnx)}$$

The *N* variable represents the prescale factor (1, 8, 64, 256, or 1024).

As for the Normal mode of operation, the TOV0 Flag is set in the same timer clock cycle that the counter counts from MAX to 0x00.

Timer 0 Fast PWM Mode

In fast PWM mode, the compare unit allows generation of PWM waveforms on the OC0x pins. Setting the COM0x1:0 bits to two will produce a non-inverted PWM and an inverted PWM output can be generated by setting the COM0x1:0 to three: Setting the COM0A1:0 bits to one allows the OC0A pin to toggle on Compare Matches if the WGM02 bit is set. This option is not available for the OC0B pin (see Table 16-3 on page 126). The actual OC0x value will only be visible on the port pin if the data direction for the port pin is set as output. The PWM waveform is generated by setting (or clearing) the OC0x Register at the Compare Match between OCR0x and TCNT0, and clearing (or setting) the OC0x Register at the timer clock cycle the counter is cleared (changes from TOP to BOTTOM).

The PWM frequency for the output can be calculated by the following equation:

$$f_{OCnxPWM} = \frac{f_{clk\_I/O}}{N \cdot 256}$$

The *N* variable represents the prescale factor (1, 8, 64, 256, or 1024).

Table 16-3. Compare Output Mode, Fast PWM Mode[1]

COM0A1	COM0A0	Description
0	0	Normal port operation, OC0A disconnected
0	1	WGM02 = 0: Normal Port Operation, OC0A Disconnected WGM02 = 1: Toggle OC0A on Compare Match
1	0	Clear OC0A on Compare Match, set OC0A at BOTTOM (non-inverting mode)
1	1	Set OC0A on Compare Match, clear OC0A at BOTTOM (inverting mode)

Timer 1

Table 17-2. Waveform Generation Mode Bit Description[1]

Mode	WGMn3	WGMn2 (CTCn)	WGMn1 (PWMn1)	WGMn0 (PWMn0)	Timer/Counter Mode of Operation	TOP	Update of OCRnx at	TOVn 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	OCRnA	Immediate	MAX
5	0	1	0	1	Fast PWM, 8-bit	0x00FF	BOTTOM	TOP
6	0	1	1	0	Fast PWM, 9-bit	0x01FF	BOTTOM	TOP
7	0	1	1	1	Fast PWM, 10-bit	0x03FF	BOTTOM	TOP
8	1	0	0	0	PWM, Phase and Frequency Correct	ICRn	BOTTOM	BOTTOM
9	1	0	0	1	PWM,Phase and Frequency Correct	OCRnA	BOTTOM	BOTTOM
10	1	0	1	0	PWM, Phase Correct	ICRn	TOP	BOTTOM
11	1	0	1	1	PWM, Phase Correct	OCRnA	TOP	BOTTOM
12	1	1	0	0	CTC	ICRn	Immediate	MAX
13	1	1	0	1	(Reserved)	–	–	–
14	1	1	1	0	Fast PWM	ICRn	BOTTOM	TOP
15	1	1	1	1	Fast PWM	OCRnA	BOTTOM	TOP

Timer 1 Fast PWM Mode

In fast PWM mode, the compare units allow generation of PWM waveforms on the OCnx pins. Setting the COMnx1:0 bits to two will produce a non-inverted PWM and an inverted PWM output can be generated by setting the COMnx1:0 to three (see Table on page 155). The actual OCnx value will only be visible on the port pin if the data direction for the port pin is set as output (DDR_OCnx). The PWM waveform is generated by setting (or clearing) the OCnx Register at the compare match between OCRnx and TCNTn, and clearing (or setting) the OCnx Register at the timer clock cycle the counter is cleared (changes from TOP to BOTTOM).

The PWM frequency for the output can be calculated by the following equation:

$$f_{OCnxPWM} = \frac{f_{clk\_I/O}}{N \cdot (1 + TOP)}$$

The N variable represents the prescaler divider (1, 8, 64, 256, or 1024).

26.8.3 ADCSRA – ADC Control and Status Register A

Bit	7	6	5	4	3	2	1	0	
(0x7A)	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
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	

- **Bit 7 – ADEN: ADC Enable**

Writing this bit to one enables the ADC. By writing it to zero, the ADC is turned off. Turning the ADC off while a conversion is in progress, will terminate this conversion.

- **Bit 6 – ADSC: ADC Start Conversion**

In Single Conversion mode, write this bit to one to start each conversion. In Free Running mode, write this bit to one to start the first conversion. The first conversion after ADSC has been written after the ADC has been enabled, or if ADSC is written at the same time as the ADC is enabled, will take 25 ADC clock cycles instead of the normal 13. This first conversion performs initialization of the ADC.

ADSC will read as one as long as a conversion is in progress. When the conversion is complete, it returns to zero. Writing zero to this bit has no effect.

- **Bit 5 – ADATE: ADC Auto Trigger Enable**

When this bit is written to one, Auto Triggering of the ADC is enabled. The ADC will start a conversion on a positive edge of the selected trigger signal. The trigger source is selected by setting the ADC Trigger Select bits, ADTS in ADCSRB.

- **Bit 4 – ADIF: ADC Interrupt Flag**

This bit is set when an ADC conversion completes and the Data Registers are updated. The ADC Conversion Complete Interrupt is executed if the ADIE bit and the I-bit in SREG are set. ADIF is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, ADIF is cleared by writing a logical one to the flag. Beware that if doing a Read-Modify-Write on ADCSRA, a pending interrupt can be disabled. This also applies if the SBI and CBI instructions are used.

- **Bit 3 – ADIE: ADC Interrupt Enable**

When this bit is written to one and the I-bit in SREG is set, the ADC Conversion Complete Interrupt is activated.

- **Bits 2:0 – ADPS2:0: ADC Prescaler Select Bits**

These bits determine the division factor between the XTAL frequency and the input clock to the ADC.

Table 26-5. ADC Prescaler Selections

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

Figure 26-3. ADC Prescaler

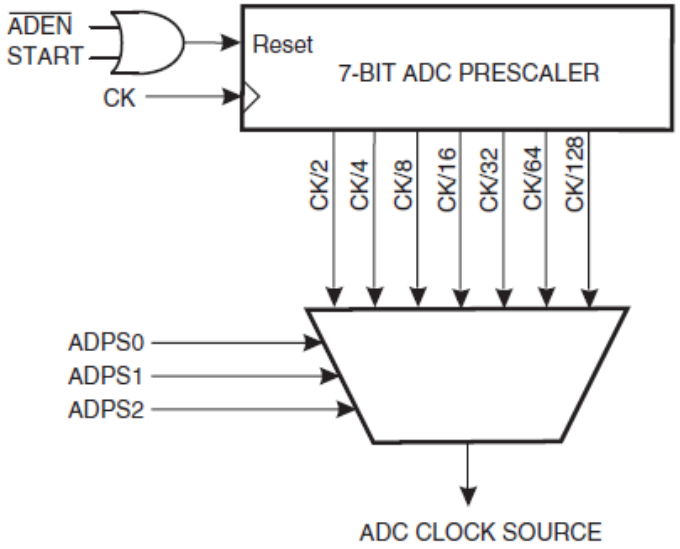


Figure 26-5. ADC Timing Diagram, Single Conversion

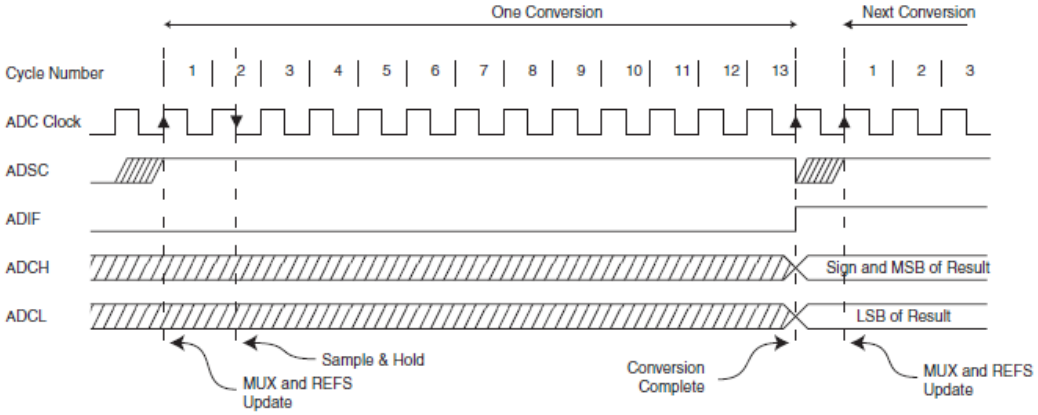


Table 26-1. ADC Conversion Time

Condition	Sample & Hold (Cycles from Start of Conversion)	Conversion Time (Cycles)
First conversion	13.5	25
Normal conversions, single ended	1.5	13
Auto Triggered conversions	2	13.5
Normal conversions, differential	1.5/2.5	13/14

Table 22-1. Equations for Calculating Baud Rate Register Setting

Operating Mode	Equation for Calculating Baud Rate[1]	Equation for Calculating UBRR Value
Asynchronous Normal mode (U2Xn = 0)	$BAUD = \frac{f_{OSC}}{16(UBRRn+1)}$	$UBRRn = \frac{f_{OSC}}{16BAUD} - 1$
Asynchronous Double Speed mode (U2Xn = 1)	$BAUD = \frac{f_{OSC}}{8(UBRRn+1)}$	$UBRRn = \frac{f_{OSC}}{8BAUD} - 1$
Synchronous Master mode	$BAUD = \frac{f_{OSC}}{2(UBRRn+1)}$	$UBRRn = \frac{f_{OSC}}{2BAUD} - 1$

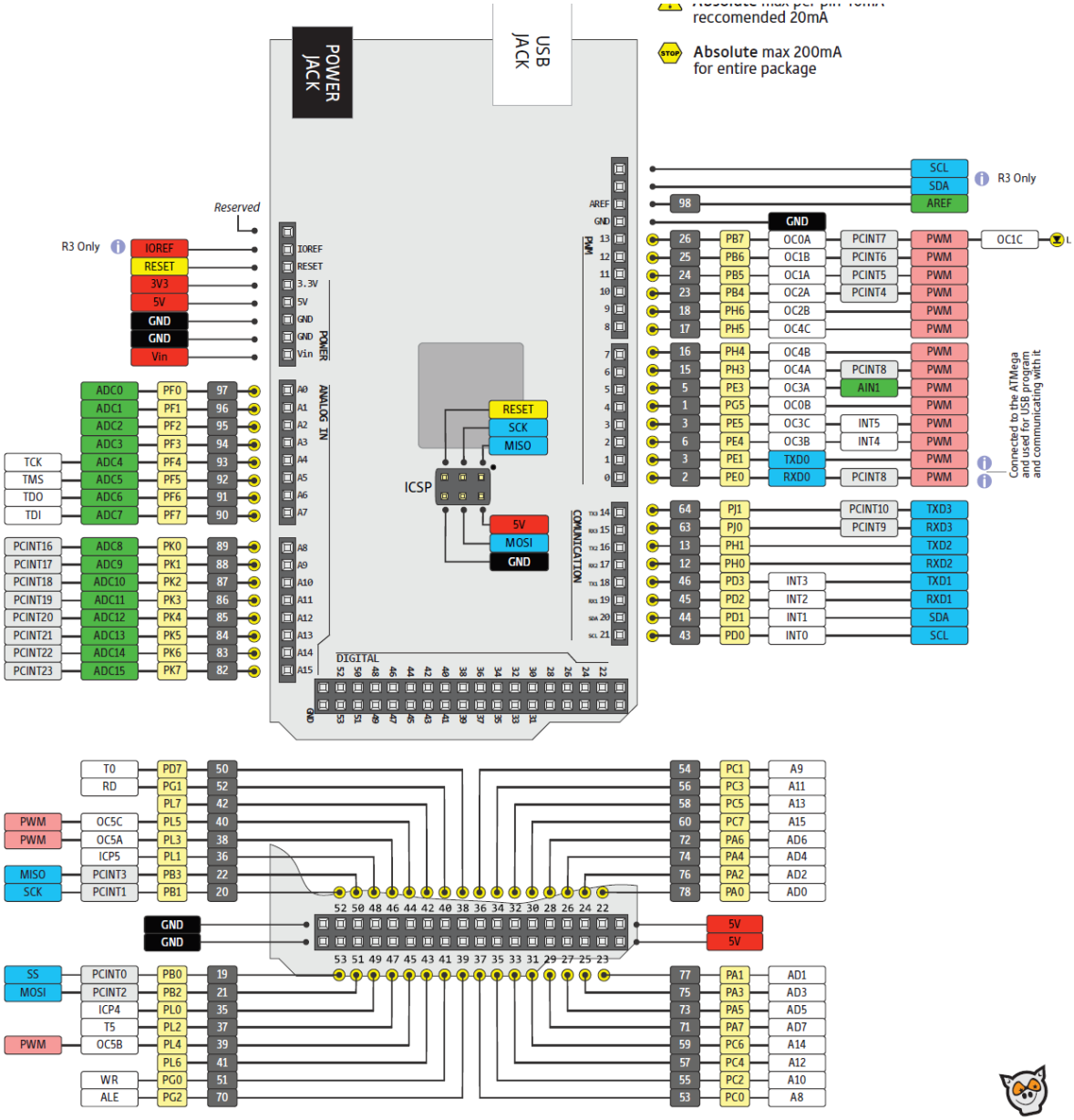
Note: 1. The baud rate is defined to be the transfer rate in bit per second (bps).

BAUD Baud rate (in bits per second, bps).

f_{osc} System Oscillator clock frequency.

UBRRn Contents of the UBRRHn and UBRRLn Registers, (0-4095).

BINARY NUMBER				HEX
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	A
1	0	1	1	B
1	1	0	0	C
1	1	0	1	D
1	1	1	0	E
1	1	1	1	F



30 JAN 2013