[This question paper contains 8 printed pages.]

Your Roll No.....

Sr. No. of Question Paper : 1180 **A**

Unique Paper Code : 32517916

Name of the Paper : Embedded Systems (DSE)

Name of the Course : **B.Sc. (H) Electronics**

Semester : VI

Duration : 3½ Hours Maximum Marks : 75

Instructions for Candidates

1. Write your Roll No. on the top immediately on receipt of this question paper.
2. There are **seven** questions in all. Attempt any **Five** questions.
3. **All** questions carry equal marks.

1. (a) What are Big Endian and Little Endian Processors? (4)

   (b) What are the different types of embedded systems based on performance and functional requirements? Briefly explain each one of them. (6)

P.T.O.

(c) What are the five difficulties inherent to building and programming an embedded system? (5)

2. (a) Give the comparison between RISC and CISC Architectures. (6)

(b) Draw a simplified block diagram of I/O port in AVR. Explain the function of DDR, PIN and PORT registers and also give the purpose of pull-up resistors. How are pull up resistors enabled in ATmega32 microcontrollers. Write the initialization code for enabling pullup resistor for all 8 pins of port A. (6)

(c) What is EEPROM? How is it different from Flash Memory. (3)

3. (a) Explain Program Counter. What is the highest ROM memory location address possible in AVR microcontrollers and why? (4)

(b) With the help of a diagram explain the various clock sources in Atmega32 Microcontroller. (5)

(c) Assume that RAM locations $90 - $94 have a string of ASCII data, as shown below

$90 = ('H') $91 = ('E') $92 = ('L') $93 = ('L') $94 = ('O')

Write a program to get each character and send it to PORTB one byte at a time. Write the program using

(i) Direct Addressing Mode

(ii) Register Indirect Addressing Mode

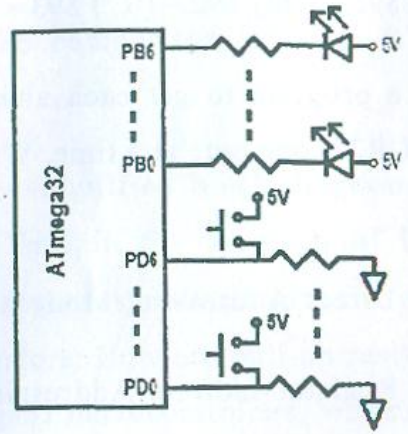
(iii) Register Indirect Addressing Mode with auto increment

Explain the program execution. (2+3+1)

4. (a) Explain CALL and RET instructions and the role of stack in their execution. (5)

(b) 7 LEDs and 7 Keys are connected to Port B and Port D as shown below. Write a program to continuously read the status of keys connected to PD0..PD6 pins. If a key is pressed, corresponding

LED on Port B should be turned ON else it should be turned OFF. Explain the program execution. (4)



(c) Indicate the value loaded into the registers in the following code snippet (6)

```
.EQU  C1 = 63

.EDQ  C2 = 0X6F

.EQU  C3 = $27

LDI R20, (C1 & C2) | (C3^C2)

LDI R21, C2 - ((C1 + C3)>>PB3)

LDI R30, (OXFB <<4) + 101
```


5. (a) What are different modes of Timer 0 in ATmega32 microcontroller? Explain Normal Mode and CTC Mode with relevant diagrams. (5)

(b) Write a subroutine to initialize Timer0 to generate a delay of 10ms. Assume clock frequency = 4 MHz.

(i) Normal Mode and

(ii) CTC Mode (5)

(c) Assume XTAL = 8 MHz. Find the clock period of Timer 0 if a prescaler option of 1024 is chosen. Also find the largest time delay that can be obtained using this prescaler. What should be the value of CS02, CS01 and CS00 bits to select the prescaler of 1024? What is the role of Waveform Generation Mode Bits (WGM01, WGM00) in TCCRO register? (5)

6. (a) What is an Interrupt? Give its advantages over Polling. Explain with an example. (5)

(b) What is interrupt latency and why does it exists? Give the vector locations and interrupt priority for INTO, INT1 and INT2. (5)

(c) Assume that the INTO pin is connected to a switch that is normally high. Write a program to initialize the interrupt and interrupt service routine (ISR) that increments the content of PORTC once by 2 whenever INTO goes low while at the same time transferring data from PORTA to PORTB. Explain the program execution. (5)

7. (a) Explain SPI serial communication bus. How is it different from USART and TWI bus? (5)

(b) Write a code snippet to initialize UART to transmit and receive data at 19200bps with 8 data bits, odd parity and 1 stop bit. Assume system clock to be 3.6864MHz. (5)

(c) Explain the following terms used in ADC

(i) Conversion Time

(ii) Step Size

(iii) Reference Voltage

(iv) Auto triggered conversion

(v) Gain Error (5)

Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
$3F ($5F)	SREG	I	T	H	S	V	N	Z	C
$3E ($5E)	SPH	–	–	–	–	SP11	SP10	SP9	SP8
$3D ($5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0
$3C ($5C)	OCR0	Timer/Counter0 Output Compare Register							
$3B ($5B)	GICR	INT1	INT0	INT2	–	–	–	IVSEL	IVCE
$3A ($5A)	GIFR	INTF1	INTF0	INTF2	–	–	–	–	–
$39 ($59)	TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0
$38 ($58)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0
$37 ($57)	SPMCR	SPMIE	RWWSB	–	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN
$36 ($56)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE
$35 ($55)	MCUCR	SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00
$34 ($54)	MCUCSR	JTD	ISC2	–	JTRF	WDRF	BORF	EXTRF	PORF
$33 ($53)	TCCR0	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00
$32 ($52)	TCNT0	Timer/Counter0 (8 Bits)							
$31[1] ($51)[1]	OSCCAL	Oscillator Calibration Register							
	OCDR	On-Chip Debug Register							
$30 ($50)	SFIOR	ADTS2	ADTS1	ADTS0	–	ACME	PUD	PSR2	PSR10
$2F ($4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10
$2E ($4E)	TCCR1B	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10
$2D ($4D)	TCNT1H	Timer/Counter1 – Counter Register High Byte							
$2C ($4C)	TCNT1L	Timer/Counter1 – Counter Register Low Byte							
$2B ($4B)	OCR1AH	Timer/Counter1 – Output Compare Register A High Byte							
$2A ($4A)	OCR1AL	Timer/Counter1 – Output Compare Register A Low Byte							
$29 ($49)	OCR1BH	Timer/Counter1 – Output Compare Register B High Byte							
$28 ($48)	OCR1BL	Timer/Counter1 – Output Compare Register B Low Byte							
$27 ($47)	ICR1H	Timer/Counter1 – Input Capture Register High Byte							
$26 ($46)	ICR1L	Timer/Counter1 – Input Capture Register Low Byte							
$25 ($45)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20
$24 ($44)	TCNT2	Timer/Counter2 (8 Bits)							
$23 ($43)	OCR2	Timer/Counter2 Output Compare Register							
$22 ($42)	ASSR	–	–	–	–	AS2	TCN2UB	OCR2UB	TCR2UB
$21 ($41)	WDTCR	–	–	–	WDTOE	WDE	WDP2	WDP1	WDP0
$20[2] ($40)[2]	UBRRH	URSEL	–	–	–	UBRR[11:8]			
	UCSRC	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL
$1F ($3F)	EEARH	–	–	–	–	–	–	EEAR9	EEAR8
$1E ($3E)	EEARL	EEPROM Address Register Low Byte							
$1D ($3D)	EEDR	EEPROM Data Register							
$1C ($3C)	EECR	–	–	–	–	EERIE	EEMWE	EEWE	EERE
$1B ($3B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0
$1A ($3A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0
$19 ($39)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0
$18 ($38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
$17 ($37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0
$16 ($36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0
$15 ($35)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0
$14 ($34)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
$13 ($33)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0
$12 ($32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0
$11 ($31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0
$10 ($30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0
$0F ($2F)	SPDR	SPI Data Register							
$0E ($2E)	SPSR	SPIF	WCOL	–	–	–	–	–	SPI2X
$0D ($2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
$0C ($2C)	UDR	USART I/O Data Register							
$0B ($2B)	UCSRA	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM
$0A ($2A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8
$09 ($29)	UBRRL	USART Baud Rate Register Low Byte							
$08 ($28)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0
$07 ($27)	ADMUX	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0
$06 ($26)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0
$05 ($25)	ADCH	ADC Data Register Hi Byte							
$04 ($24)	ADCL	ADC Data Register Low Byte							
$03 ($23)	TWDR	Two-wire Serial Interface Data Register							
$02 ($22)	TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
$01 ($21)	TWSR	TWS7	TWS6	TWS5	TWS4	TWS3	–	TWPS1	TWPS0
$00 ($20)	TWBR	Two-wire Serial Interface Bit Rate Register							