Scheme of Evaluation
(Scoring indicators)

Revision : 2015
code:5041 **Course**
Course Title:Embedded systems

Qst No	Scoring Indicator	Split up score	Sub Total	Total
I	**PART A**			
1.	8 bit RISC,Harvard architecture,On chip ROM,RAM,EEPROM,TIMERS ,I/O PORTS.ADC,PWM,USART,SPI,I2C,CAN,USB etc(Any two)	2		
2.	.MACRO name ……. …….. .ENDMACRO	2	10	
3.	RESET	2		
4.	Kernal manages all the tasks to achieve the desired performance of the embedded system.	2		
5.	Real time,Non real time,mobile or any two examples	2		

Scoring Indicator

PART B

II

(1)

Family	Part no.	ROM	RAM	EEPROM	I/O PINS	TIMERS
Classic	AT90s2313	2K	128 BYTE	128 BYTE	15	2
Mega	ATmega32	32	2K	1K	32	4
Tiny	ATtiny13	1K	64 BYTE	64byte	6	1
Special purpose	AT90CAN128	128K	4K	4K	5	4

Any three family with any three comparisons only

3+3 | 6

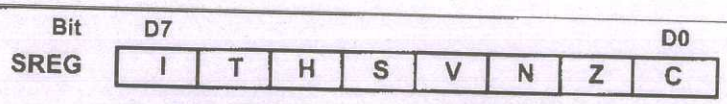
(2)

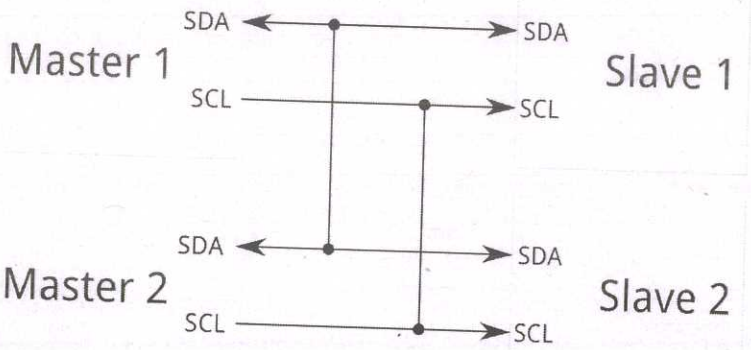
Step 1: Load initial timer count (TCNTX register)
Step2: Enable timer interrupt bit in TIMSK register
Step3:Enable interrutp Globally(SEI)

Description only

3x2 | 6

(3)	LDI R16,0X12
	LDI R17,0X04
	MUL R16,R17
	STS 0X300,R0
	STS 0X301,R1
	HERE:SJMP HERE (any addressing mode can be used)
		6	6	

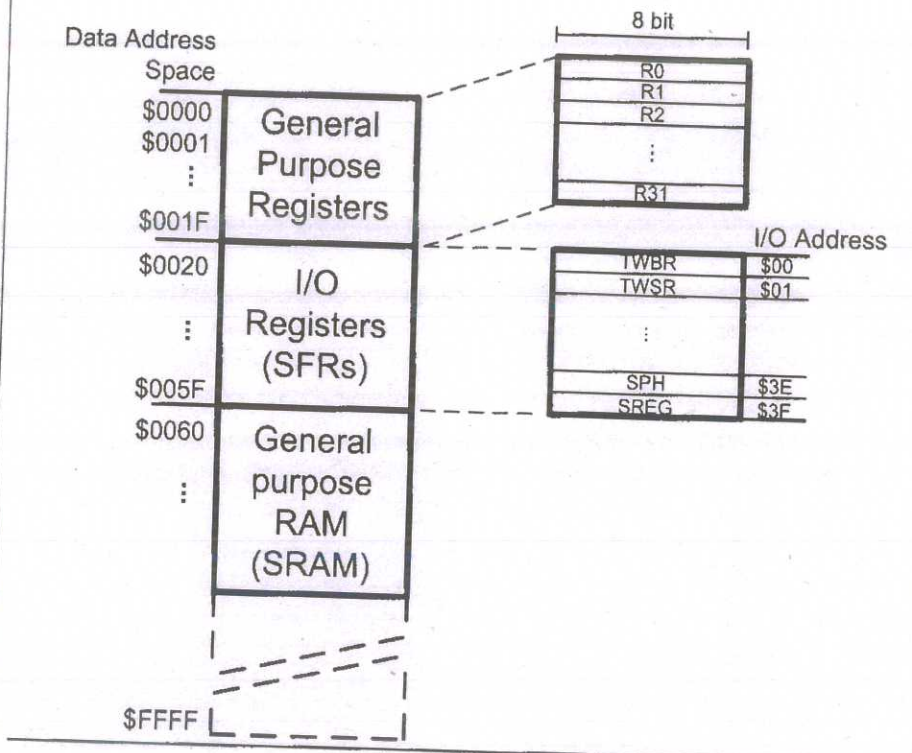
4	***Consumer application** eg:DVD player,microwave oven,remote controller. ***Industrial automation** eg:pharmaceutical,cement,suger etc ***Medical electronics** eg:EEG,ECG,BP measurement ***Computer networking** eg:bridges,routers,ISDN etc ***Telecommunications** eg: Key telephone,terminal adaptor,web cameras ***Wireless technologies** eg:mobile phone,palmtops etc ***Instrumentation** eg:Oscilloscope,spectrum analyzer,measuring instruments ***Security** eg:security systems at homes ,offices,airports etc ***Finance** eg: ATM,SMART cards Any six with one example for each	6	6	
5	 C – Carry flag Z – Zero flag N – Negative flag V – Overflow flag S – Sign flag H – Half carry T – Bit copy storage I – Global Interrupt Enable	3+3	6	

6	Delay program consists of two parts Setting a counter and a loop The count value and the clock frequency decieds the delay Example: DELAY: LDI R20,0XFF 1 AGAIN: NOP 1 NOP 1 DEC R20 1 BRNE AGAIN 2/1 RET TAIME DELAY=(1+(255x5)-1+4)x1μs=1279μs 1μs is the clock time period.	1x6	6	
7	 **(Description)**	4+2	6	

III

PART C

a)



4+4 | 8

15

b)

Explanation

Feature	8052	Pic 18F452	ATmega32
Program ROM	**8K**	**32K**	**32K**
Data RAM	**256 byte**	**2K**	**2K**
EEPROM	**0**	**256 bytes**	**1K**
Timers	**3**	**4**	**3**
I/O pins	**32**	**35**	**32**

(Comparison of any two micocontroller with ATmega 32) | 7 | 7

IV

a)

1. Single-Register (Immediate)
2. Register
3. Direct
4. Register indirect
5. Flash Direct
6. Flash Indirect

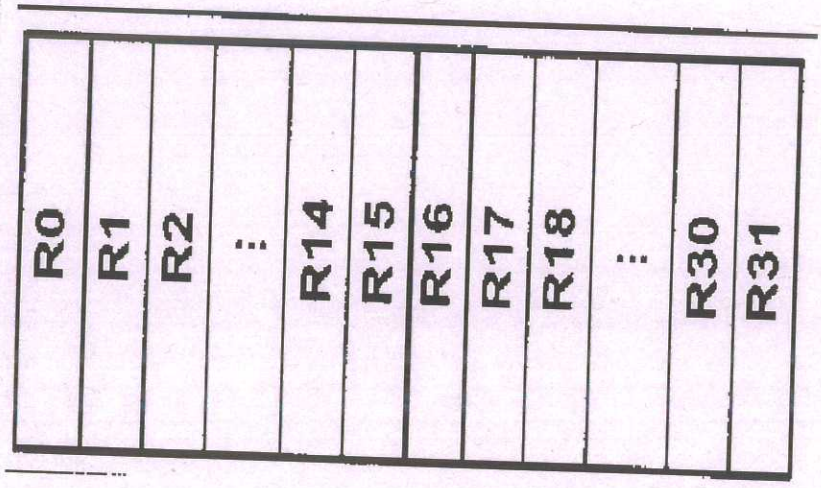
Description

b)

***32 GPRS**
***R0 TO R31**
***Located at lowest address of RAM**
***Adress from 0x00 to 0x1F**
***All are 8 bits**

4+4

8



15

3+4

7

Description

V (a)	Assembler directives are the pseudo instructions which gives directions to the assembler. [labe:] mnemonic [operands] [;comment] Any example program ….	2+3 +3	8	

(b)	LDI R20,HIGH(RAMEND)
	OUT SPH,R20
	LDI R20,LOW(RAMEND)
	OUT SPL,R20
	
	SBI DDRB,3
	HERE :SBI PORTB,3
	CALL DELAY
	CBI PORTB,3
	CALL DELAY
	RJMP HERE
		7	7	15

VI

(a)

Instruction	Action
BRLO	Branch if C=1
BRSH	Branch if C=0
BREQ	Branch if Z=1
BRNE	Branch if Z=0
BRMI	Branch if N=1
BRPL	Branch if N=0
BRVS	Branch if V=1
BRVC	Branch if V=0

Any four with description — 4x2 — 8

(b) — 7

```
        CBI DDRB,2
        LDI R16,0XFF
        OUT DDRC,R16
AGAIN:  SBIS PINB ,2
        RJMP AGAIN
        LDI R16,0X45
        OUT PORTC,R16
HERE:   RJMP HERE
```

7

VII

(a)

Data Type	Size in Bits
unsigned char	8-bit
char	8-bit
unsigned int	16-bit
int	16-bit
unsigned long	32-bit
long	32-bit
float	32-bit
double	32-bit

4+4

Small description

8

(b)

```
#include<avr/io.h>
int main (void)
{
DDRB=0X00;
DDRC=0XFF;
Unsigned char temp ;
While(1)
{
Temp = PINB;
PORTC = temp;
}
}
```

7 7

VII

I

(a)

1. Load the TCNT0 register with the initial count value.
2. Load the value into the TCCR0 register, indicating which mode (8-bit or 16-bit) is to be used and the prescaler option. When you select the clock source, the timer/counter starts to count, and each tick causes the content of the timer/counter to increment by 1.
3. Keep monitoring the timer overflow flag (TOV0) to see if it is raised. Get out of the loop when TOV0 becomes high.
4. Stop the timer by disconnecting the clock source, using the following instructions:

```
LDI   R20,0x00
OUT   TCCRC,R20   ;timer stopped, mode=Normal
```

5. Clear the TOV0 flag for the next round.
6. Go back to Step 1 to load TCNT0 again.

8 | 8

b)

AVR ATmega32 has three external hardware interrupts

They are connected on pins PD2, PD3 and PB2

Referred as INT0, INT1 and INT2 respectively.

Upon activation of these interrupts, the ATmega controller gets interrupted in whatever task it is doing and jumps to perform the interrupt service routine.

External interrupts can be level triggered or edge triggered.

We can program this triggering. INT0 and INT1 can be level triggered and edge triggered

Where as INT2 can be only edge triggered.

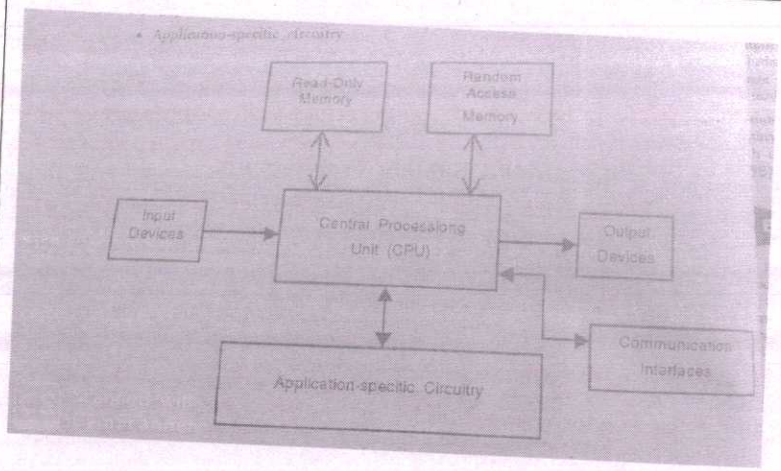
Vector address of INT0,INT1,INT2 are 0002,0004,0006 respectively

(Any seven points only)

7 | 7

IX

(a)



4+4

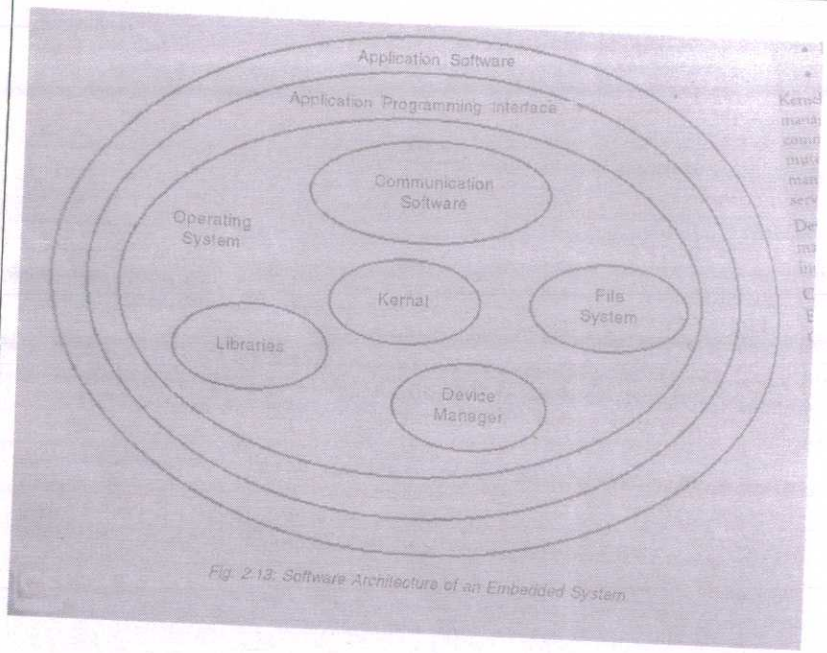
Explanation

8

(b) Embedded systems do a very specific task.
They have very limited resources.
They have to work against some dead lines
They are constrained for power.
They are need to be highly reliable.
They have to operate in extreme environmental conditions .
Embedded systems that address the consumer market are very cost sensitive

Any seven

7 7

X					
(a)	 Fig. 2.13: Software Architecture of an Embedded System		4+4	8	
	Explanation				
(b)	Non real time embedded operating system Eg:Embedded Linux,Embedded NT and Windows xp embedded Real time operating system Eg:QNX Neutrino,VxWorks,RTLinux,MICRO C/OSII and OS9 Mobile/hand held operating system Eg:Palm OS,Symbian OS ,Windows CE		7	7	