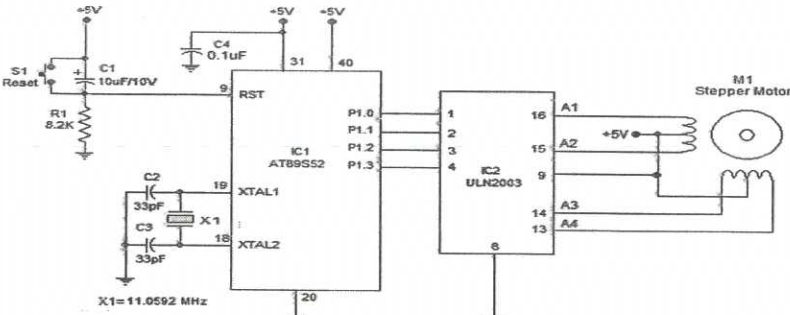


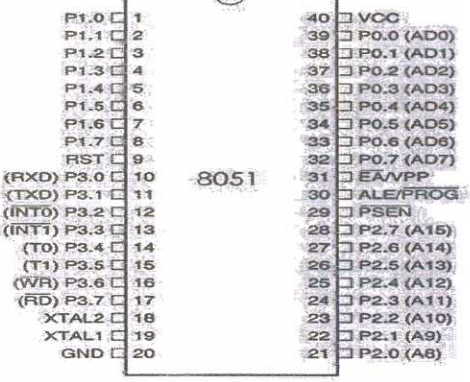
Qst.No.	Scoring Indicator	Split up score	Sub total	Tota																					
I	PART A																								
1	8-bit CPU,4K bytes of on-chip ROM,128-bytes of on chip RAM,32 bi-directional I/O lines, two 16-bit timers.	2																							
2	CJNE= Compare and jump if not equal. Eg : CJNE A, #data, rel DJNZ= Decrement and jump if register is not equal to zero.eg: DJNZ Rn,rel	2																							
3	Rate of information communication. Baud rates are available to vary the speed of data transfer. The standard rates are 300,600,1200,2400,4800,9600 and 19200.	2																							
4	Register select(RS),Enable(E),Read/Write(R/W).	2																							
5	Interfacing with 8051 microcontroller can be defined as transferring data from interfacing peripherals such as sensors, motors, machines, etc to 8051 and vice versa. By interfacing with 8051, we can have ease of control over complex circuit components or devices.	2		10																					
II	PART B																								
II 1.	<table border="1"> <thead> <tr> <th>Feature</th> <th>8031</th> <th>8051</th> </tr> </thead> <tbody> <tr> <td>1. ROM(on chip)in bytes</td> <td>0K</td> <td>4K</td> </tr> <tr> <td>2. RAM, in bytes</td> <td>128</td> <td>128</td> </tr> <tr> <td>3.Timers</td> <td>2</td> <td>2</td> </tr> <tr> <td>4.I/O pins</td> <td>32</td> <td>32</td> </tr> <tr> <td>5.Serial port</td> <td>1</td> <td>1</td> </tr> <tr> <td>6.Interrupt sources</td> <td>6</td> <td>6</td> </tr> </tbody> </table>	Feature	8031	8051	1. ROM(on chip)in bytes	0K	4K	2. RAM, in bytes	128	128	3.Timers	2	2	4.I/O pins	32	32	5.Serial port	1	1	6.Interrupt sources	6	6	6x1	6	6
Feature	8031	8051																							
1. ROM(on chip)in bytes	0K	4K																							
2. RAM, in bytes	128	128																							
3.Timers	2	2																							
4.I/O pins	32	32																							
5.Serial port	1	1																							
6.Interrupt sources	6	6																							
II 2.		3																							

Qst.No.	Scoring Indicator	Split up score	Sub total	Total
	<p>Port-0 can be configured as a normal bidirectional I/O port or it can be used for address/data interfacing for accessing external memory. When control is '1', the port is used for address/data interfacing. When the control is '0', the port can be used as a normal bidirectional I/O port. Let us assume that control is '0'. When the port is used as an input port, '1' is written to the latch. In this situation both the output MOSFETs are 'off'. Hence the output pin floats. This high impedance pin can be pulled up or low by an external source. When the port is used as an output port, a '1' written to the latch again turns 'off' both the output MOSFETs and causes the output pin to float.</p>	3	6	6
II 3	<pre>MOV A,#data1 MOV B,#data2 MUL AB MOV DPTR,#4600 MOVX @DPTR,A HERE: SJMP HERE</pre>	6	6	6
II 4	<p>LJMP (long jump):-LJMP is an unconditional long jump. It is a 3-byte instruction in which the first byte is the opcode, and the second and third bytes represent the 16-bit address of the target location. The 2-byte target address allows a jump to any memory location from 0000 to FFFFH.</p> <p>SJMP (short jump):-In this 2-byte instruction, the first byte is the opcode and the second byte is the relative address of the target location. The relative address range of 00 to FFH is divided into forward and backward jumps; that is, within -128 to +127 bytes of memory relative to the address of the current PC (program counter). If the jump is forward, the target address can be within a space of 127 bytes from the current PC. If the target address is backward, the target address can be within -128 bytes from the current PC.</p>	3		
II 5	<p>Mode select bits of TMOD register: The M1 and M0 are mode select bits, which are used to select the timer operations. There are four modes to operate the timers.</p> <p>Mode 0: This is a 13-bit mode that means the timer operation completes with "8192" pulses.</p> <p>Mode 1: This is a 16-bit mode, which means the timer operation completes with maximum clock pulses that "65535".</p> <p>Mode 2: This mode is an 8-bit auto-reload mode, which means the timer operation completes with only "256" clock pulses.</p> <p>Mode 3: This mode is a split-timer mode, which means the loading values in T0 and automatically starts the T1.</p>	3	6	6
		4x1.5	6	6

Qst.No.	Scoring Indicator	Split up score	Sub total	Total																												
II 6	<table border="1" data-bbox="300 271 1161 459"> <thead> <tr> <th>M1</th> <th>M0</th> <th>MODE</th> <th>DESCRIPTION</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Mode 0</td> <td>13 bit timer</td> </tr> <tr> <td>0</td> <td>1</td> <td>Mode 1</td> <td>16 bit timer</td> </tr> <tr> <td>1</td> <td>0</td> <td>Mode 2</td> <td>8 bit auto reload</td> </tr> <tr> <td>1</td> <td>1</td> <td>Mode 3</td> <td>Split timer</td> </tr> </tbody> </table> <div data-bbox="204 555 1190 613" style="background-color: #cccccc; text-align: center; padding: 5px;"> <h3 style="margin: 0;">TCON Register</h3> </div> <p data-bbox="204 613 839 651">TCON : Timer/Counter Control Register (Bit Addressable)</p> <table border="1" data-bbox="325 667 1070 707" style="width: 100%; text-align: center;"> <tr> <td>TF1</td> <td>TR1</td> <td>TF0</td> <td>TR0</td> <td>IE1</td> <td>IT1</td> <td>IE0</td> <td>IT0</td> </tr> </table> <p data-bbox="204 734 1190 1153"> TF1 TCON.7 Timer 1 overflow flag. Set by hardware when the Timer/Counter 1 overflows. Cleared by hardware as processor vectors to the interrupt service routine. TR1 TCON.6 Timer 1 run control bit. Set/cleared by software to turn Timer/Counter ON/OFF. TF0 TCON.5 Timer 0 overflow flag. Set by hardware when the Timer/Counter 0 overflows. Cleared by hardware as processor vectors to the service routine. TR0 TCON.4 Timer 0 run control bit. Set/cleared by software to turn Timer/Counter 0 ON/OFF. IE1 TCON.3 External Interrupt 1 edge flag. Set by hardware when External Interrupt edge is detected. Cleared by hardware when interrupt is processed. IT1 TCON.2 Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt. IE0 TCON.1 External Interrupt 0 edge flag. Set by hardware when External Interrupt edge detected. Cleared by hardware when interrupt is processed. IT0 TCON.0 Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt. </p>	M1	M0	MODE	DESCRIPTION	0	0	Mode 0	13 bit timer	0	1	Mode 1	16 bit timer	1	0	Mode 2	8 bit auto reload	1	1	Mode 3	Split timer	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	2	2	6
M1	M0	MODE	DESCRIPTION																													
0	0	Mode 0	13 bit timer																													
0	1	Mode 1	16 bit timer																													
1	0	Mode 2	8 bit auto reload																													
1	1	Mode 3	Split timer																													
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0																									
II 7	 <p data-bbox="287 1512 1204 1736">The circuit diagram for interfacing stepper motor to 8051 is shown above. P1.0, P1.1, P1.2 and P1.3 pins are used for controlling the phases A1, A2, A3 and A4 of the stepper motor respectively. ULN2003 is used for driving the individual phases of the stepper motor. ULN2003 is a darlington transistor array used for driving high current loads such as relays and motors</p>	4	6	6																												

Qst.No.	Scoring Indicator	Split up score	Sub total	Total
III a	<p>It is an 8-bit microcontroller with 4kb of ROM storage and 128 bytes of RAM storage, 2 16-bit timers. It consists of are four parallel 8-bit ports, which are programmable as well as addressable as per the requirement. An on-chip crystal oscillator is integrated in the microcontroller.</p> <p>Central Processor Unit (CPU):- It monitors and controls all operations that are performed on the Microcontroller units.</p> <p>Memory:- 8051 microcontroller has 4K of code memory or program memory,that has 4KB ROM and also 128 bytes of data memory of RAM</p> <p>Input/Output Port:- 8051 has 4 input, output ports to connect it to the other peripherals</p> <p>Timers/Counters:- 8051 microcontroller has two 16 bit timers and counters.</p> <p>Interrupts:- Generally five interrupt sources are there in 8051 Microcontroller. There are 5 vectored interrupts are shown in below INTO,TFO,INT1,TF1,RI/TI. Out of these, $(INT0)^-$ and $(INT1)^-$ are external interrupts.</p> <p>BUS:Address Bus: Microcontroller 8051 has a 16 bit address bus for transferring the data. It is used to address memory locations and to transfer the address from CPU to Memory of the microcontroller. Data Bus: Microcontroller 8051 has 8 bits of the data bus, which is used to carry data of particular applications.</p> <p>Oscillator:- microcontroller 8051 has an on-chip oscillator which works as a clock source for Central Processing Unit of the microcontroller.</p>	5	9	

Qst.No.	Scoring Indicator	Split up score	Sub total	Total																
III b	<table border="1" style="width:100%; border-collapse: collapse;"> <thead> <tr> <th style="width:50%; text-align: center;">Microprocessor</th> <th style="width:50%; text-align: center;">Microcontroller</th> </tr> </thead> <tbody> <tr> <td>1. Contains only CPU, RAM, ROM, I/O, timer are separately interfaced.</td> <td>1. CPU, RAM, ROM, I/O and timer are all on a single chip</td> </tr> <tr> <td>2. Designer decides on the amount of ROM, RAM and I/O ports.</td> <td>2. Fix amount of on-chip ROM, RAM, I/O ports</td> </tr> <tr> <td>3. High cost</td> <td>3. Low cost</td> </tr> <tr> <td>4. General-purpose</td> <td>4. Single-purpose</td> </tr> <tr> <td>5. High speed</td> <td>5. Low speed</td> </tr> <tr> <td>6. Higher Power Consumption</td> <td>6. Low Power Consumption</td> </tr> <tr> <td>ex – 8085, 8086</td> <td>ex – 8051, AVR</td> </tr> </tbody> </table>	Microprocessor	Microcontroller	1. Contains only CPU, RAM, ROM, I/O, timer are separately interfaced.	1. CPU, RAM, ROM, I/O and timer are all on a single chip	2. Designer decides on the amount of ROM, RAM and I/O ports.	2. Fix amount of on-chip ROM, RAM, I/O ports	3. High cost	3. Low cost	4. General-purpose	4. Single-purpose	5. High speed	5. Low speed	6. Higher Power Consumption	6. Low Power Consumption	ex – 8085, 8086	ex – 8051, AVR	3+3	6	15
Microprocessor	Microcontroller																			
1. Contains only CPU, RAM, ROM, I/O, timer are separately interfaced.	1. CPU, RAM, ROM, I/O and timer are all on a single chip																			
2. Designer decides on the amount of ROM, RAM and I/O ports.	2. Fix amount of on-chip ROM, RAM, I/O ports																			
3. High cost	3. Low cost																			
4. General-purpose	4. Single-purpose																			
5. High speed	5. Low speed																			
6. Higher Power Consumption	6. Low Power Consumption																			
ex – 8085, 8086	ex – 8051, AVR																			
IV a	<p style="text-align: center;">INTERNAL RAM</p> <p>8051 Microcontroller have 256B of RAM. In this 256B, the first 128B i.e. memory addresses from 00H to 7FH is divided in to Working Registers (organized as Register Banks), Bit – Addressable Area and General Purpose RAM (also known as Scratchpad area).In the first 128B of RAM (from 00H to 7FH), the first 32B i.e. memory from addresses 00H to 1FH consists of 32 Working Registers that are organized as four banks with 8 Registers in each Bank. The 4 banks are named as Bank0, Bank1, Bank2 and Bank3. Each Bank consists of 8 registers named as R0 – R7. The next 16B of the RAM i.e. from 20H to 2FH are Bit – Addressable memory locations. The final 80B of the internal RAM i.e. addresses from 30H to 7FH, is the general purpose RAM area which are byte addressable.</p>	4	8																	

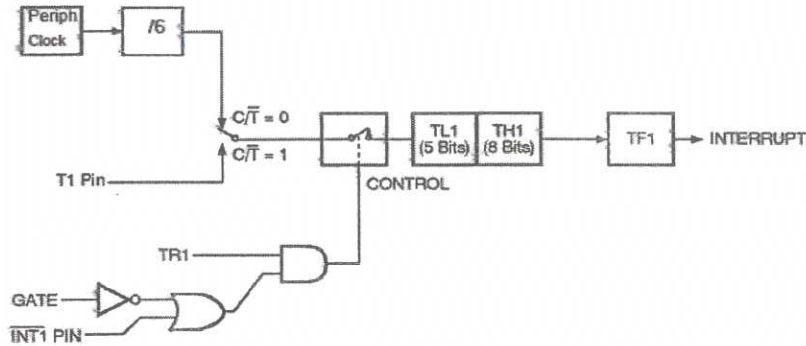
Qst.No.	Scoring Indicator	Split up score	Sub total	Total
IV b	 <p>40 - PIN DIP</p> <ul style="list-style-type: none"> • PSEN pin which stands for Program Store Enable. It is used to read a signal from the external program memory. • EA pin which stands for External Access input. It is used to enable/disable the external memory interfacing. • ALE pin which stands for Address Latch Enable. It is used to demultiplex the address-data signal of port. 	4		
V a	<ul style="list-style-type: none"> • Immediate Addressing Mode:-In this Mode, the data is provided in the instruction itself. eg:- MOV A,#0AH • Register addressing mode:- In this mode the source or destination data should be present in a register (R0 to R7).eg:-MOV A,R5 • Direct Addressing Mode:- In this mode source or destination address is specified by using 8-bit data in the instruction. Only the internal data memory can be used in this mode.eg:- MOV R2,45 • Register indirect addressing mode :- In this mode, the source or destination address is given in the register. By using register indirect addressing mode, the internal or external addresses can be accessed. The R0 and R1 are used for 8-bit addresses, and DPTR is used for 16-bit addresses, no other registers can be used for addressing purposes.eg:- MOV A,@R0 • indexed addressing mode:- In this mode source memory can only be accessed from program memory only. The destination operand is always the register A.eg:- MOVC A,@A+DPTR 	3	7	15
		5x2	10	

Qst.No.	Scoring Indicator	Split up score	Sub total	Total
V b	1.Timer 0 overflow interrupt- TF0 2.Timer 1 overflow interrupt- TF1 3.External hardware interrupt- INT0 4.External hardware interrupt- INT1 5.Serial-communication interrupt- RI/TI The Timer and Serial interrupts are internally generated by the microcontroller, whereas the external interrupts are generated by additional <u>interfacing devices</u> or switches that are externally connected to the microcontroller. These external interrupts can be edge triggered or level triggered. When an interrupt occurs, the microcontroller executes the interrupt service routine so that memory location corresponds to the interrupt that enables it.	5x1	5	15
VI a	Upon activation of the interrupt, the microcontroller goes through the following steps; 1.It finishes the instruction it is executing and saves the address of the next instruction on the stack. 2.It jumps to interrupt vector table that holds the address of the ISR 3. The microcontroller gets address of the ISR from interrupt vector table and jumps to it. It starts to execute the ISR until it reaches RET instruction. 4.Upon executing the RETI instruction, the microcontroller goes to the place where it was first interrupted and starts to execute from that address	7		
VI b	<pre> MOV R0,#60H MOV DPTR,#4500 MOV R2,#0AH L1: MOV A,@R0 MOVX @DPTR,A INC R0 INC DPTR DJNZ R2,L1 HERE:SJMP HERE </pre>	8		15

Qst.No.	Scoring Indicator	Split up score	Sub total	Total																																								
VII a	<p>The following steps must be taken to receive data serially:-</p> <ol style="list-style-type: none"> 1. TMOD register is loaded with 20H, indicating Timer1, mode 2 2. TH1 is loaded with a value of required Baud Rate 3. SCON register is loaded with the value 50H, indicating serial mode 1 4. TR1 is set to 1 to start Timer 1 5. RI is cleared with instruction 'CLR RI' 6. Monitor RI flag bit using the instruction 'JNB RI, xx' 7. When RI is raised, SBUF has the byte and its contents are moved into a safe place. 8. To receive next character go to step 5 																																											
VII b	<p>TMOD : Timer/Counter Mode Control Register (Not Bit Addressable)</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 20px;">GATE</td> <td style="width: 20px;">C/T</td> <td style="width: 20px;">M1</td> <td style="width: 20px;">M0</td> <td style="width: 20px;">GATE</td> <td style="width: 20px;">C/T</td> <td style="width: 20px;">M1</td> <td style="width: 20px;">M0</td> </tr> <tr> <td colspan="4" style="text-align: center;">TIMER 1</td> <td colspan="4" style="text-align: center;">TIMER 0</td> </tr> </table> <p>GATE When TR_x (in TCON) is set and GATE = 1, TIMER/COUNTER_x will run only while INT_x pin is high (hardware control). When GATE = 0, TIMER/COUNTER_x will run only while TR_x = 1 (software control).</p> <p>C/T Timer or Counter selector. Cleared for Timer operation (input from internal system clock). Set for Counter operation (input from Tx input pin).</p> <p>M1 Mode selector bit (NOTE 1).</p> <p>M0 Mode selector bit (NOTE 1).</p> <p>Note 1 :</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>M1</th> <th>M0</th> <th colspan="2">OPERATING MODE</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>13-bit Timer</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>16-bit Timer/Counter</td> </tr> <tr> <td>1</td> <td>0</td> <td>2</td> <td>8-bit Auto-Reload Timer/Counter</td> </tr> <tr> <td>1</td> <td>1</td> <td>3</td> <td>(Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits, TH0 is an 8-bit Timer and is controlled by Timer 1 control bits.</td> </tr> <tr> <td>1</td> <td>1</td> <td>3</td> <td>(Timer 1) Timer/Counter 1 stopped.</td> </tr> </tbody> </table>	GATE	C/T	M1	M0	GATE	C/T	M1	M0	TIMER 1				TIMER 0				M1	M0	OPERATING MODE		0	0	0	13-bit Timer	0	1	1	16-bit Timer/Counter	1	0	2	8-bit Auto-Reload Timer/Counter	1	1	3	(Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits, TH0 is an 8-bit Timer and is controlled by Timer 1 control bits.	1	1	3	(Timer 1) Timer/Counter 1 stopped.	7		
GATE	C/T	M1	M0	GATE	C/T	M1	M0																																					
TIMER 1				TIMER 0																																								
M1	M0	OPERATING MODE																																										
0	0	0	13-bit Timer																																									
0	1	1	16-bit Timer/Counter																																									
1	0	2	8-bit Auto-Reload Timer/Counter																																									
1	1	3	(Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits, TH0 is an 8-bit Timer and is controlled by Timer 1 control bits.																																									
1	1	3	(Timer 1) Timer/Counter 1 stopped.																																									
VIII a	<p>In this mode timer operates as 13 bit up-counter the lower 5-bits are in the TL1 and the upper 8-bits are in the TH1 register. The timer interrupt flag TF1 bit is set when the timer rolls over from FFH to 00H.</p>	8	15																																									
		3																																										

Qst.No.	Scoring Indicator	Split up score	Sub total	Total
---------	-------------------	----------------	-----------	-------

Figure 2-13. Timer/Counter 1 Mode 0: 13-bit Counter



VIII b

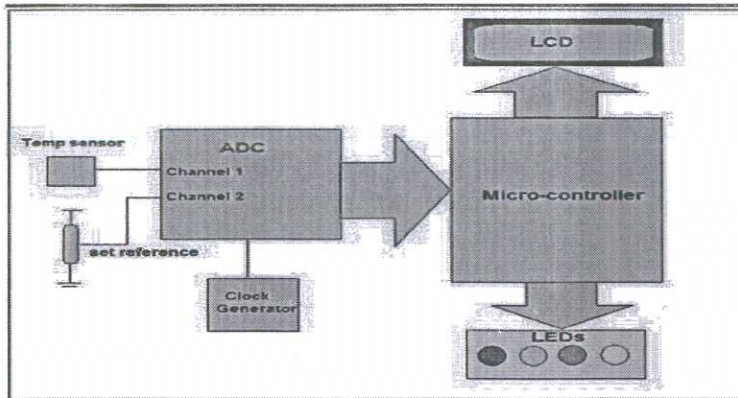
The SCON Register

MSB		LSB					
SM0	SM1	SM2	REN	TB8	RB8	TI	RI
Bit	Name	Description					
SCON.7	SM0	Serial Port Mode bit 0					
SCON.6	SM1	Serial Port Mode bit 1					
SCON.5	SM2	Multiprocessor Communication Enable					
SCON.4	REN	Receive Enable Set to enable reception. CLR to disable reception.					
SCON.3	TB8	Bit 8 of message to transmit Used to transmit optional parity bit					
SCON.2	RB8	Bit 8 of received message Receives optional parity bit					
SCON.1	TI	Transmit Interrupt Flag Set when Byte in SBUF is completely transmitted.					
SCON.0	RI	Receive Interrupt Flag Set when a valid byte is received into SBUF					

Embedded Systems Design

8

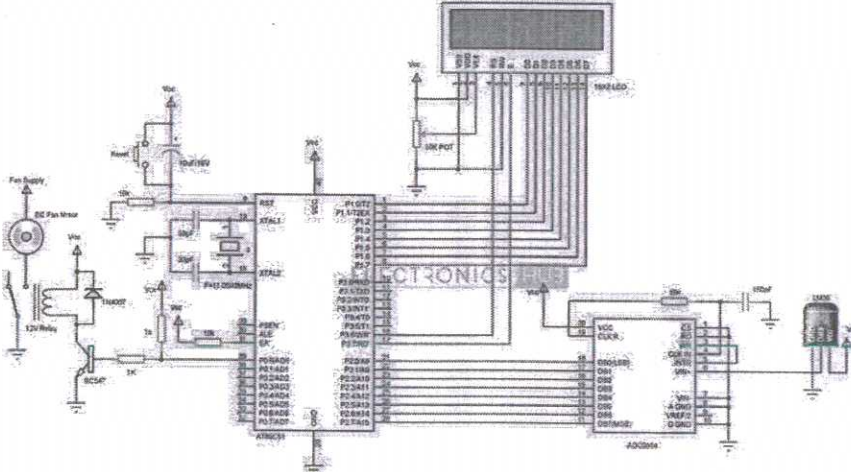
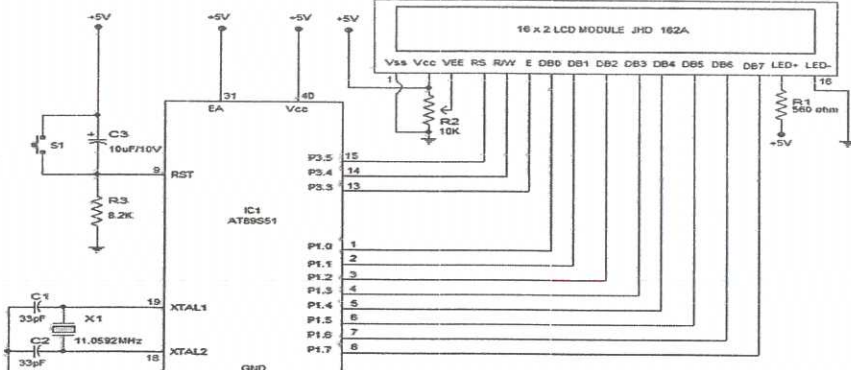
IX a



As shown in above figure, major building blocks of system are temperature sensor, Analog to Digital Converter (ADC), micro-controller, LCD, clock generator and LED indicators.

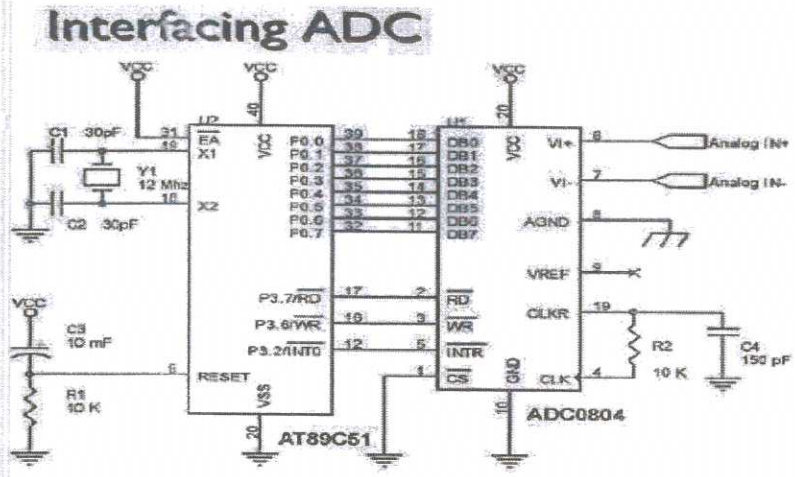
Temperature sensor: It's a transducer. It gives corresponding voltage (or current) output as change in temperature. It can be calibrated to degree Celsius. Otherwise it has to be calibrated first. **Reference potentiometer:** It sets reference temperature between min to max value. The system operation depends upon this set temperature value.

4

Qst.No.	Scoring Indicator	Split up score	Sub total	Total
	<p>ADC: Its analog to digital converter with built in multiplexer. It takes two analog inputs one from temperature sensor and another from reference potentiometer. It gives 8-bit digital output corresponding selected analog input. To get the digital output of any one channel, micro controller will select the required channel and takes digital output.</p> <p>Clock generator: ADC requires clock signal for its operation. This clock signal is generated by IC555 based clock generator.</p> <p>Micro controller: it controls operation of ADC and LCD. It takes digital output of both channels and displays them on LCD. It takes suitable decision by comparing two temperatures. Also it gives different indications on LEDs</p> 	4	8	
IX b	 <p style="text-align: center;">Interfacing 16x2 LCD module to 8051 www.circuitstoday.com Capacitor</p>	4		
	<p>C3, resistor R3 and push button switch S1 forms the reset circuitry. Ceramic capacitors C1,C2 and crystal X1 is related to the clock circuitry which produces the system clock frequency. P1.0 to P1.7 pins of the microcontroller is connected to the DB0 to DB7 pins of the module respectively and through this route the data goes to the LCD module. P3.3, P3.4 and P3.5 are connected to the E, RW, RS pins of the microcontroller and through this route the control signals are transferred to the LCD module. Resistor R1 limits the current through the back light LED and so do the back light intensity. POT R2 is used for adjusting the contrast of the display</p>	3	7	15

Qst.No.	Scoring Indicator	Split up score	Sub total	Tota
---------	-------------------	----------------	-----------	------

X a

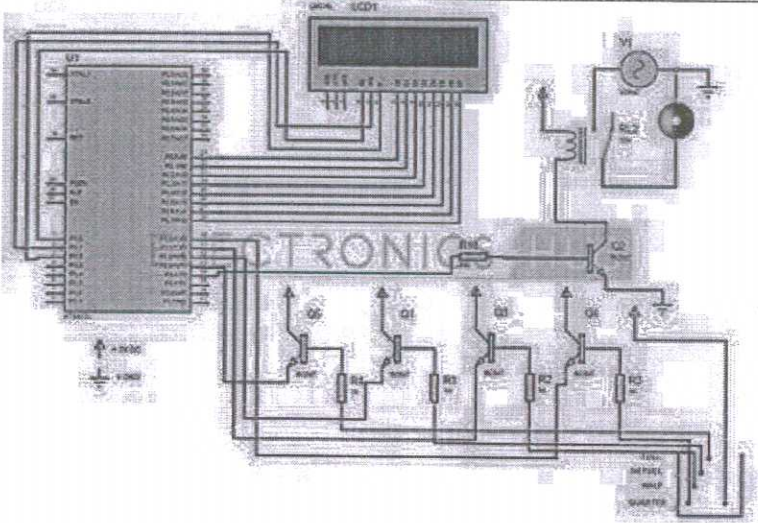


The figure above shows the schematic for interfacing ADC0804 to 8051. The circuit initiates the ADC to convert a given analogue input , then accepts the corresponding digital data and displays it on the LED array connected at P0.. Data out pins (D0 to D7) of the ADC0804 are connected to the port pins P1.0 to P1.7 respectively. LEDs D1 to D8 are connected to the port pins P0.0 to P0.7 respectively. Resistors R1 to R8 are current limiting resistors. In simple words P1 of the microcontroller is the input port and P0 is the output port. Control signals for the ADC (INTR, WR, RD and CS) are available at port pins P3.4 to P3.7 respectively. Resistor R9 and capacitor C1 are associated with the internal clock circuitry of the ADC. Preset resistor R10 forms a voltage divider which can be used to apply a particular input analogue voltage to the ADC. Push button S1, resistor R11 and capacitor C4 forms a debouncing reset mechanism. Crystal X1 and capacitors C2,C3 are associated with the clock circuitry of the microcontroller

4

3

7

Qst.No.	Scoring Indicator	Split up score	Sub total	Total
X b	 <p>The diagram shows a microcontroller (U1) interfaced with an LCD (LCD1) and a motor. The microcontroller has several pins connected to the LCD and the motor's control circuit. The motor is connected to a power source through a relay and a fuse. The system is designed to monitor water levels and automatically switch the motor ON when the tank is empty and OFF when the tank is full. The LCD displays the water level status: EMPTY, QUARTER, HALF, and FULL.</p> <p>This system monitors the level of water in a tank and automatically switches ON the motor whenever the water tank is empty. The motor is switched OFF when the overhead tank or container is FULL. Here, the water level of the tank is indicated on LCD (Liquid crystal Display).</p> <p>Initially when tank is empty, LCD will display the message EMPTY and motor runs automatically. When water level reaches to quarter level, now LCD displays QUARTER and still motor runs. For further levels, LCD displays the messages HALF and $\frac{3}{4}$ FULL.</p> <p>When tank is full, LCD displays FULL and motor automatically stops. Again motor runs when tank is empty.</p>	5	8	15