Introduction to Embedded Systems-IA-2

1. **What is a Watchdog timer?**
   - It is a special purpose hardware that protects the system from software hangs.
   - Watchdog timer always counts down from some large number to zero.
   - This process takes a few seconds to reset, in the meantime, it is possible for embedded software to "kick" the watchdog timer, to reset its counter to the original large number.
   - If the timer expires i.e. counter reaches zero, the watchdog timer will assume that the system has entered a state of software hang, then resets the embedded processor and restarts the software.
   - It is a common way to recover from unexpected software hangs.
   - The figure below diagrammatically represents the working of the watchdog Timer:

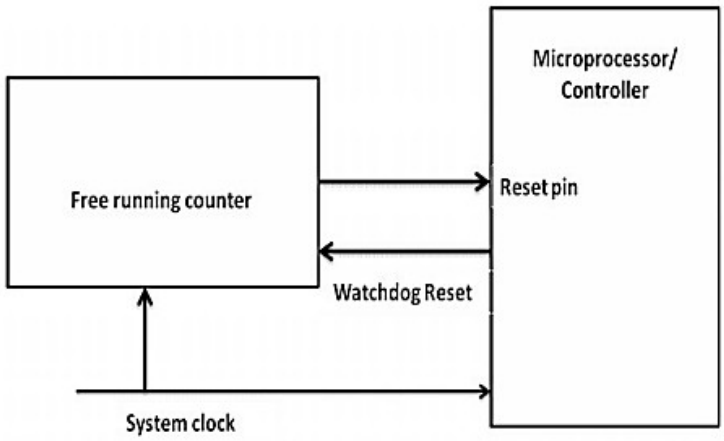
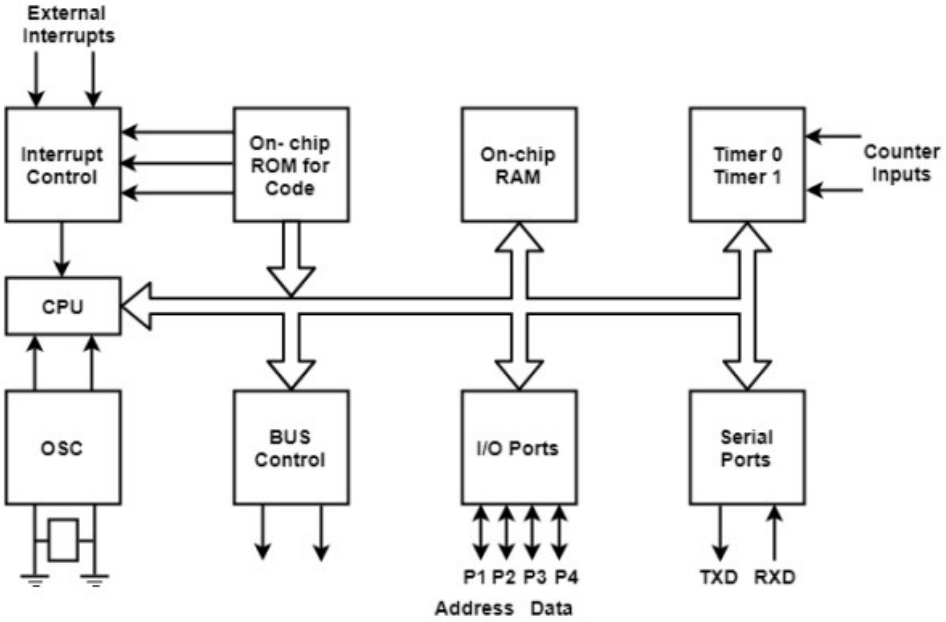


Figure: Watchdog Timer

2. **Explain 8051 Microcontroller Hardware.**



   - Central Processor Unit (CPU)-The CPU is the brain of any processing device of the microcontroller. It monitors and controls all operations that are performed on the Microcontroller unit

- Interrupts-Interrupt is a subroutine call that interrupts the microcontrollers main operations or work and causes it to execute any other program, which is more important at the time of operation.
- Memory- The memory which is used to store the program of the microcontroller is known as code memory or Program memory of applications. It is known as ROM memory of microcontroller. The data memory of the 8051 is used to store data temporarily for operation and is known as RAM memory.
- Bus-Bus is a collection of wires, which work as a communication channel or medium for transfer of Data. These buses consist of 8, 16 or more wires of the microcontroller. Thus, these can carry 8 bits, 16 bits simultaneously. Two types of buses:
    - 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.
- Oscillator: Oscillator works as a clock source for Central Processing Unit of the microcontroller. The output pulses of oscillator are stable. Therefore, it enables synchronized work of all parts of the 8051 Microcontroller.
- Input / Output Port: Microcontroller 8051 has four input, output ports to connect it to the other peripherals.
- Timers/Counters: The 8051 microcontroller has two 16-bit timers and counters. The timers are used for measurement of intervals to determine the pulse width of pulses.

3. Explain the following:

a. Checksum.

b. CRC

- Checksum:
    - To check whether the data stored in a memory device is valid or not, the checksum of the data in the memory device is computed and stored along with the data.
    - The moment when we have to confirm the validity of the data, we just have to recalculate the checksum and compare it with previous checksum. If the two checksums match, the data is assumed to be valid.
    - The simplest checksum algorithm is to add up all the data bytes discarding carries.
    - A Checksum is usually stored at some fixed location in memory. This makes it easy to compute and store the check sum for the very first time and later on to compare the recomputed checksum with the original one.
    - Disadvantage: A simple sum-of-data checksum cannot detect many of the most common data errors.
- CRC – Cyclic Redundancy Check:
    - A Cyclic Redundancy Check is a specific checksum algorithm designed to detect the most common data errors.
    - CRCs are frequently used in Embedded Applications that requires the storage or transmission of large blocks of data.
    - The CRC works as follows:

- A division operation occurs between the message at numerator and the generator polynomial at denominator. The generator polynomial is a fixed smaller length binary string.
- The remainder of the division operation is the CRC Checksum

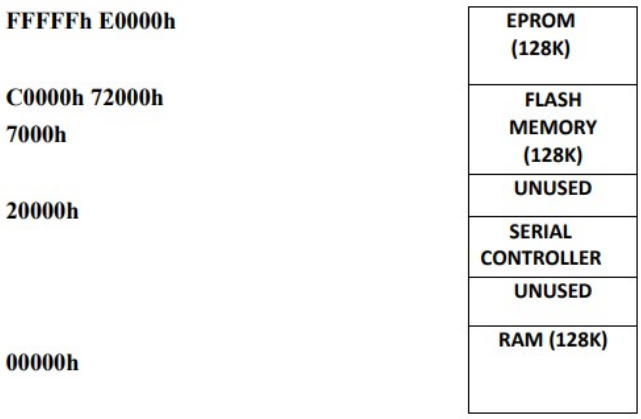
4. Explain components of Device Driver.

A device driver can be implemented (as components) in the following steps:

- A data structure that overlays the memory-mapped control and status registers of the device: This basic step involves creating a C style structure that is actually a map of the registers present in the device. These registers can be found out by referring to the data sheet for the device. A table is created which maps the control register to their relative offsets.
- A set of variables to track the current state of the hardware and device driver: It involves listing out the required variables needed to keep track of the state of the hardware and device driver.
- Initialize the hardware: Once the variables to be used are known the next step in device driver programming is to initialize the hardware.
- A set of routines that provide an API for users of the device driver: This involves writing different functions that will implement the various tasks listed to be performed by the device.
- Interrupt service routines: Once the required functions and routines are coded the thing remaining to be done is to identify and write routines for servicing the interrupts.

5. What is a Memory map?

- A Memory Map is the processor's "address book." It shows what these devices look like to the processor.
- The memory map contains one entry for each of the memories and peripherals that are accessible from the processor's memory space.
- All processors store their programs and data in memory. These chips are located in the processor's memory space, and the processor communicates with them by way of two sets of electrical wires called the address bus and the data bus.
- To read or write a particular location in memory, the processor first writes the desired address onto the address bus. The data is then transferred over the data bus.
- A memory map is a table that shows the name and address range of each memory device and peripheral that is located in the memory space.
- The lowest address is at the bottom and the highest address is at the top. Each time a new device is added, add it to the memory map, it is placed in its approximate location in memory and the starting and ending addresses are labelled in hexadecimal.
- The diagram below shows the memory map for the printer sharing device:



- The part of the header file below describes the memory map

```
#define RAM_BASE (void *) 0x00000000
#define SC_BASE (void *)0x70000000
#define SC_INTACK (void *) 0x70001000
#define FLASH_BASE (void *) 0xC0000000
#define EPROM_BASE (void *) 0xE0000000
```

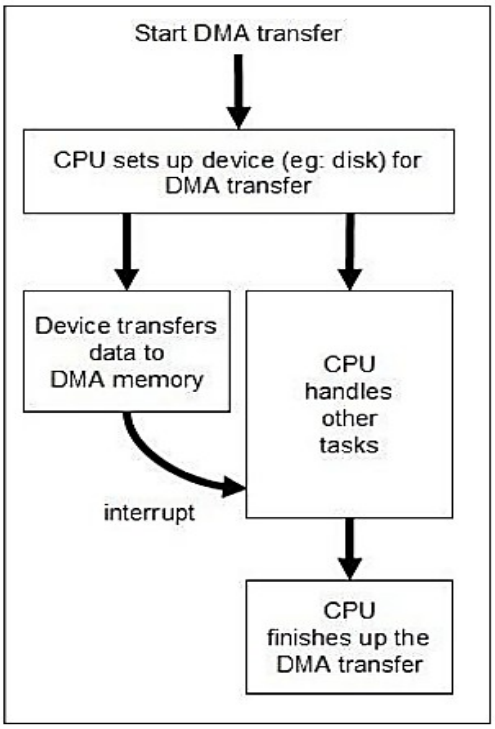
6. Explain Automotive Communication Buses.

- Controller Area Network (CAN): The CAN bus was originally proposed by Robert Bosch, pioneer in the Automotive embedded solution providers. It supports medium speed and high speed data transfer. CAN is an event-driven protocol interface with support for error handling in data transmission. It is generally employed in safety systems like airbag control; powertrain systems like engine control and Antilock Brake System (ABS); and navigation systems like GPS.
- Local Interconnect Network (LIN): The LIN bus is a single master multiple slave (up to 16 independent slave nodes) communication interface. LIN is a low speed, single wire communication interface with support for data rates up to 20 Kbps and is used for sensor/actuator interfacing. LIN bus follows the master communication triggering technique to eliminate the possible bus arbitration problem that can occur by the simultaneous talking of different slave nodes connected to a single interface bus. LIN bus is employed in applications like mirror controls, fan controls, seat positioning controls, window controls, and position controls where response time is not a critical issue.
- Media-Oriented System Transport (MOST): The Media-oriented system transport (MOST) is targeted for automotive audio/video equipment interfacing, used primarily in European cars. A MOST bus is a multimedia fibre-optic point-to-point network implemented in a star, ring or daisy-chained over optical fibre cables. The MOST bus-specifications define the physical (electrical and optical parameters) layer as well as the application layer, network layer, and media access control. MOST bus is an optical fibre cable connected between the Electrical Optical Converter (EOC) and Optical Electrical Converter (OEC), which would translate into the optical cable MOST bus.

7. What is Direct Memory Access (DMA)?

- DMA is a technique for transferring blocks of data directly between two hardware devices.
- In the absence of DMA the processor must read the data from one device and write it to the other one byte or word at a time.
- DMA Absence Disadvantage: If the amount of data to be transferred is large or frequency of transfer is high the rest of the software might never get a chance to run.

- DMA Presence Advantage: The DMA Controller performs the entire transfer with little help from the Processor.
- Working of DMA:
    - The Processor provides the DMA Controller with source and destination address & total number of bytes of the block of data which needs transfer.
    - After copying each byte each address is incremented & remaining bytes are reduced by one.
    - When the number of bytes reaches zeros the block transfer ends & DMAController sends an Interrupt to Processor.



8. What are the features of an 8051 Microcontroller?

- 8 Bit microcontroller
- 40 Pins DIP (Dual In package)
- 4KB bytes on-chip program memory (ROM)
- 128 bytes on-chip data memory (RAM)
- Four register banks
- 128 user defined software flags
- 8-bit bidirectional data bus
- 16-bit unidirectional address bus
- 32 general purpose registers each of 8-bit
- 16 bit Timers (usually 2, but may have more or less)
- Three internal and two external Interrupts
- Four 8-bit ports, (short model have two 8-bit ports)

- 16-bit program counter and data pointer
- 8051 may also have a number of special features such as UARTs, ADC, Op-amp(Operational amplifier), etc.

9. Explain the following:

a. RXD.
b. TXD.
c. INT0 and INT1.
d. T0 and T1.

- (RXD): RXD (serial data receive pin) is for serial input. Through this input signal microcontroller receives data for serial communication.
- (TXD): TXD (serial data transmit pin) is serial output pin. Through this output signal microcontroller transmits data for serial communication.
- (INT0′, INT1′): External Hardware Interrupt 0 and Interrupt 1 respectively. When this interrupt is activated (i.e. when it is low), 8051 gets interrupted in whatever it is doing and jumps to the vector value of the interrupt (0003H for INT0 and 0013H for INT1) and starts performing Interrupt Service Routine (ISR) from that vector location.
- (T0 and T1): Timer 0 and Timer 1 external input. They can be connected with a 16 bit timer/counter.

10. Explain the working of embedded systems with respect to Washing Machines.

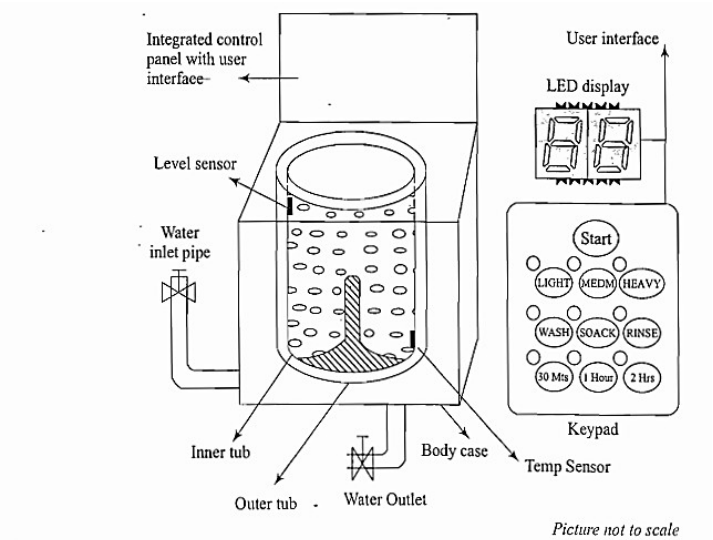


Fig. 4.2 Washing machine – Functional block diagram

- The actuator part of the washing machine consists of a motorised agitator, tumble tub, water drawing pump and inlet valve to control the flow of water into the unit.
- The sensor part consists of the water temperature sensor, level sensor, etc. The control part contains a microprocessor/ controller based board with interfaces to the sensors and actuators. The sensor data is fed back to the control unit and the control unit generate; the necessary actuator outputs.
- The control unit also provides connectivity to user interfaces like keypad for setting the washing time, selecting the type of material to be washed like light, medium, heavy duty, etc.

- Washing machine comes in two models, namely, top loading and front loading machines. In top loading models the agitator of the machine twists back and forth and pulls the cloth down to the bottom of the tub. On reaching the bottom of the tub the clothes work their way back up to the top of the tub where the agitator grabs them again and repeats the mechanism. In the front loading machines, the clothes are tumbled and plunged into the water over and over again.