

ESD Assignment

① Explain development environment in detail

Development Environment:

software and hardware tools supports the programming of general purpose processor
two processors in ES are:

i. the development processor, on which we write and debug our program. this processor is part of desktop computer

ii. the other processor is the target processor, to which we will send program and which will form part of ES's implementation

→ for example we may develop our system on a pentium processor, but use a motorola

⇒ some times the two processors are same.

⇒ Assembler translate assembly instructions to binary machine instructions

⇒ An assembler may also translate symbolic labels into actual addresses

⇒ The mapping of assembly instructions to machine instructions is one-to-one

⇒ A linker is a software tool that creates an executable file

⇒ compiler translate structured programs into machine or assembly programs

⇒ cross compilers are extremely common in embedded system development

② Explain programmers view in detail

Programmers View:-

A programmer writes the program instructions that carryout the desired functionality

⇒ may need not to know detailed information about the processor's architecture & operation

⇒ Instead may deal with architectural abstraction

⇒ the level of abstraction depends on the level of programming

The two levels of programming are

→ assembly language programming

→ structured language programming

→ assembly language programming - represents processor specific instructions

→ structured language programming - uses processor specific instructions

→ compiler automatically translates processor independent instructions into processor specific instructions.

③ explain optimizing the original program:

optimizing a program means improving its performance by making it faster, more efficiency, more efficient & more power efficient without changing what program does

Goals of optimization:

1. Reduce execution time

2. Reduce memory usage

3. Improve power efficiency

4. Increase responsiveness

ways to optimize a program:-

1. Code simplification

2. Use efficient algorithms

3. memory optimisation

4. loop optimization

5. Avoid redundancy

6. compiler optimization.

④ Explain the following - i. Register - ii. Shift register - iii. Counter.

Register: A register is a small, fast storage area inside a CPU or other digital circuit. It temporarily holds data, instructions, & addresses. Registers are used for quick read/write operations & are essential in performing arithmetic, logic & data transfer operations.

Shift register: A shift register is a type of register where the bits of data can be shifted left or right. It is used to store data & move it in or out one bit at a time. Often used for data conversion.

Types: SISO, SIPO, PISO, PIP0

Counter: A counter is a sequential circuit that counts pulses or events. It increments & decrements a binary number based on clock input & commonly used for counting, timing & controlling operations.

Types: Upcounter, Downcounter, Up/Down counter, Ring counter,

⑤ illustrate program state machine

A program state machine is a model used to present the sequence of states a program or system can be in & how it transitions from one state to state to another based on inputs or events

It is commonly used in embedded systems, control systems & software design to organize logic clearly

Illustration of a program state machine:

Traffic light controller (simple-3 state psm)

states : 1. Red
2. Green
3. Yellow

Transitions : ① Red \rightarrow Green
② Green \rightarrow Yellow
③ Yellow \rightarrow Red

Fan speed controller (using button presses)

states :- off
low
medium
high

Transitions: Each button press changes the state

off \rightarrow low \rightarrow medium \rightarrow high \rightarrow off (and cycles again)

states : Enter user name
Enter password
Authenticating
login success
login fail

Transitions :

open login page \rightarrow start \rightarrow enter user name

enter credentials \rightarrow Enter user name \rightarrow Enter password

submit \rightarrow enter password \rightarrow Authenticating

correct \rightarrow Authenticating \rightarrow login success

⑥ Elaborate concurrent process model

The concurrent process model is used in embedded systems to describe systems where multiple processes, tasks run simultaneously appear to run simultaneously, often on single & multiple processors. It helps structure & manage the parallel activities in the system.

In Embedded systems you often deal with multiple tasks happening at once such as:

- Reading sensor data
- Controlling a motor
- Communicating with other devices
- Updating a display

The concurrent process model allows for efficient multitasking.

Types of concurrency:

1. True concurrency
2. Pseudo concurrency

Elements in the model:

1. process / tasks
2. shared resources
3. Synchronisation mechanisms
4. communication

Benefits

1. Efficient multitasking
2. Real Time responsiveness
3. Scalable for complex systems.