Ans-1 ades samelance of (a) A Real time system is a system which is designed to handle events as they occur in real time. There is a great importance of speed at which tasks are completed. These tasks are due to be performed within a specific time constraint. Hard real time systems & soft real time Systems are both used in industry for different tasts. The primary difference between hard and soft real time is that their consequences of missing a deadline differ from each other RTLinux has a special design for its kernal because it has two kernals. RTLmur was a specialized seal time pernal called RTcore.

The second cernal which is used for regular applications that donot have time constraint.

Both threed handling of interrupt handly are controlled by RT core. This RT core also restricts the Linux ternel by making it make to disable interrupts to make sure It doesn't interfere with process schedulary. Real time applications can communicate with Lmux kernels through frost in first out pipes.

Vxworks uses a single microkernel to handle basic Kernel functions. Additional functions dike file sharing and networking have to be loaded from provided functionality.

This system provides About a flexibility to fit Its functionality without loosening its constraints on available memory & resources.

As both of them are real time operating system and an autonomous robots is a complex system that has ability to descide its actions on its environment from ets sensing, state in order to fill his alms. So we can prefer to use RTLinux / Vxworks for Os hardware and native robot software.

It is a very complex task but as both the RTOS contains Schedulers which are preemptine In a nature and is flexible by allowing different scheduling techniques.

Vrworks contains kernel for loreemptire multitasking inter process communication and interrupt response

So we can prefer both in hard / soft real

1 - b)We will prefer Real time Databases for the autonomous robot because It is a database system which uses got real time processing to handle workloads whose state is constantly changing. It uses ACID properties for timely execution of transactions. Here operations execute with predictable response, and with applications acceptable sevels of logical & temposal consistency of data which is required in our autonomous robot from lementation, As the complexity of this task is very high the amount of transactions to be handled by real time systems increases. So due to this we prefer real time detabases. Compared with traditional databases, real time database systmy have a feature which must satisfy time constraints associated With fransactions.

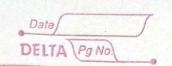
Praneactions in seal time DB systems should be scheduled considering both data consistency and timing constraints. It must adopt to changes in the operating

envisorment and guarantee the completion of critical transactions. In this database system, the correctness of transaction processing depends not only on correct results but also on the time at which transaction is completed. Bansactions must be scheduled in such a way that they can be completed before their corresponding deadlines expire In traditional DB system, research has been focused on providing high degree of proedictorbitity in task scheduling, the problem of interestry
transactors processing capabilities with timing el
consistency constraints has not received much Conventional dB systems are not used in real time applications due to two inadequancies! poor performance & lack of predictability Features of Real time Duta Base system! - Has timing constraints. - gueries to the DB should have soft & hard dealling.

- Data returned must have both absolute

consistency & relative consistency. t both seal time scheduling & database technologies can be applied on seal time data management

1-c) Due to the benefits of its remability & producting the component based approach has become the primary technology in robot sho ranworks. However all the exicting systems are very limited in fault tolerance support, even though fault tolerance function is crucial for commercial success of service robots While substantial effort has been invested in making nobots more reliable, experience demonstrates are often challenged by frequent failures Despite this, robots have not yet reached a level of design that allows effective management of faulty or unexpected be harrior of by untranel eyers Information regarding failures that occurred for robot handling in classified in a categories - Interaction Algori Hums Almorts - 80 tolked Hardware software. tions of the wall before They use several attributes to classify fautson



there properties including fault's relevance to different robotic systems -1 relevance

-1 condition

- symptons

The influence of varying reliability on total fine system and found that persons of low reliability explice during the interaction have a more negative impact on overall trut.

People trust a robot jes when reliability drops occurred late or in the middle of run.

So to handle all the faults and to increase selability we can check all the possible. test cases that might create problem while doing actual real time task.

Handling all these test cases will surely morces the reliability over the robots as a spergoon. But these will be still chances of error as may possible that some new problem arese which was not handle before.