Chapter 4

Real time software design

Definition

A real-time system is a software system where the correct functioning of the system depends on the results produced by the system and the time at which this result are produced.

- A ‘soft’ real-time system is a system groups whose operation is degraded if results are not produced according to the specified timing requirement.
- A ‘hard’ real-time system is a system whose operation is incorrect if results are not produced according to the timing specification.

Stimulus system

- Given a stimulus, system must produce a response within a specified time.
- Periodic stimulus system & stimuli which occurs at the prediction time intervals.
- Aperiodic stimuli which occur at unpredictable time interval.

General Model of Real Time System

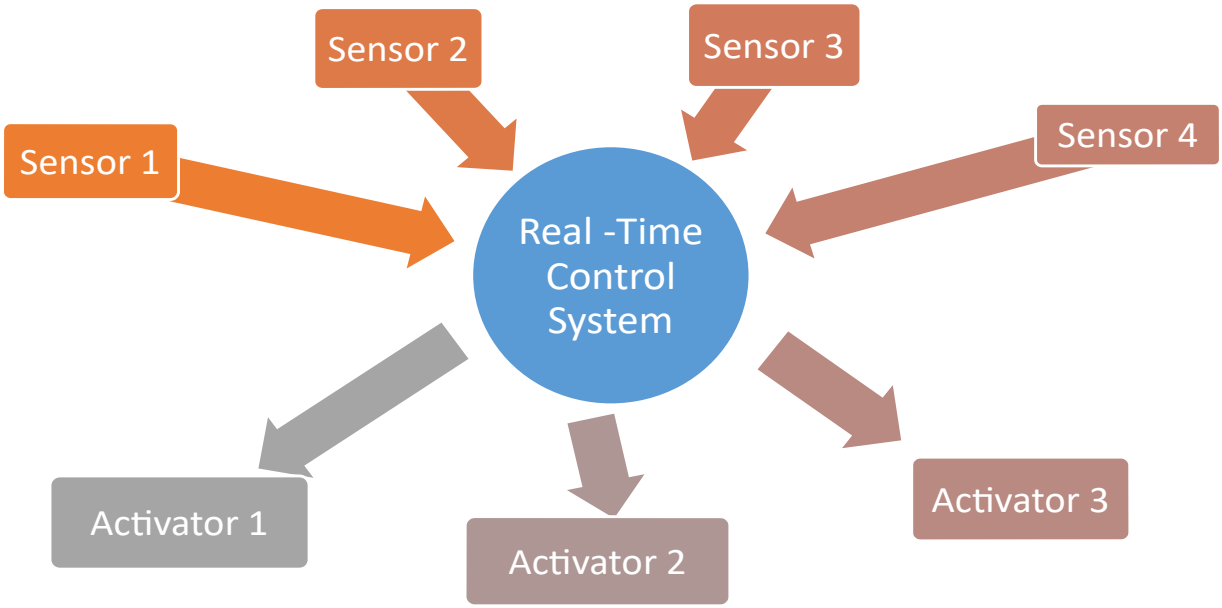

Fig : The sensor-system activator model of an embedded real-time system

Sensor/actuator process

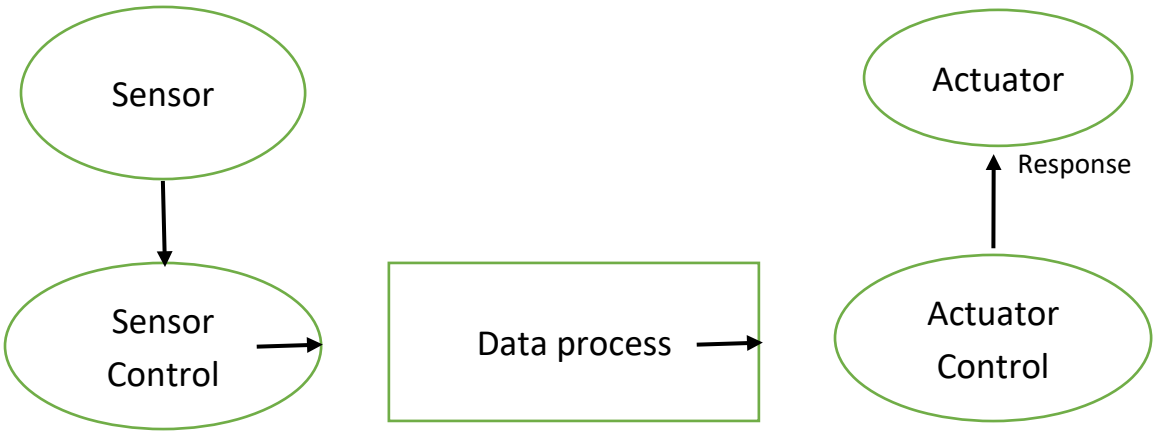

Fig : Sensor/actuator control process

- The generality of the stimulus response system of a real-time system leads to the generic architectural model where there are three types of process.

1. **Sensor control process:**
   - ✓ collect information from sensor
   - ✓ May buffer information collected in response to a sensor stimulus.
2. **Data processor:**
   - Carries out processing of collected information and compute response.
3. **Actuator**
   - Generates control signals for actuator.

4.1 System Design

- Design both the hardware & software associated with system. Partition function to either hardware of software.
- Design decision should be made on the basis of Non-functional system requirements.
- Hardware delivers better performance but potentially longer development & less scope for change.

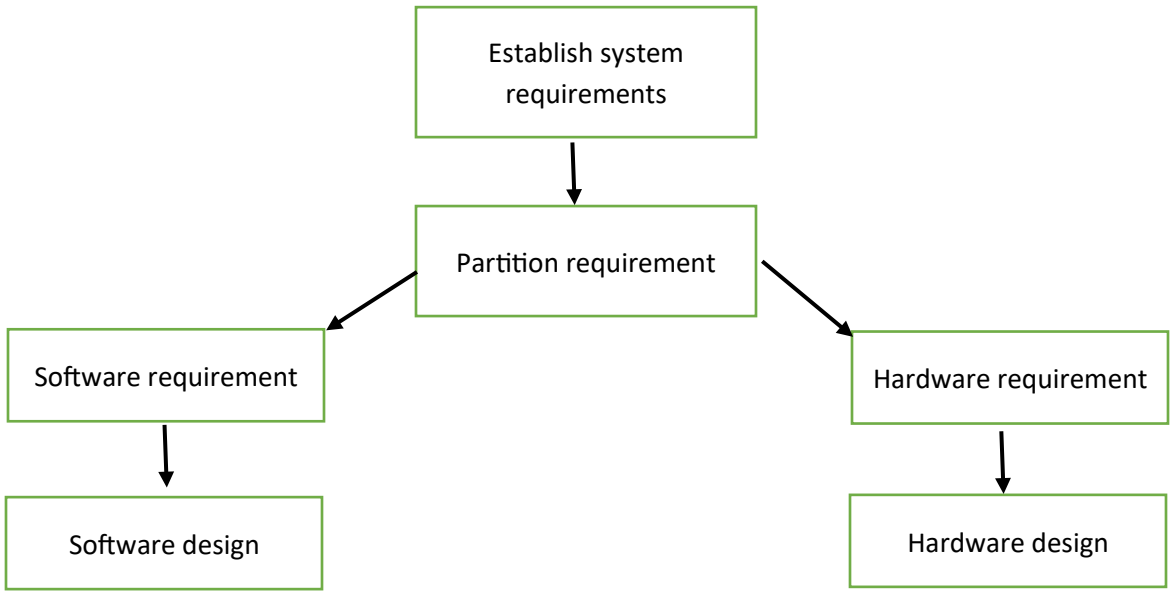

Fig: hardware and software design

- Identify the stimuli to be processed and the required responses to the stimuli.
- For each stimuli & response, identify the timing constraints
- Aggregate the stimulus & response processing into concurrent process

  A process may be associated with each class of stimulus & response.

- Design algorithms to process each class of stimulus & response. These must meet the given timing requirements.
- Design a scheduling system which will ensure that processes are started in time to meet their deadlines.
- Integrate using a real-time execution or operating system.

Timing constraints

- May require extensive simulation and experiment to ensure that they are made by the system.
- May mean that certain design strategies such as object oriented design cannot be used because of additional overhead involved.
- May mean that low level programming language features have to be used for performance reason.

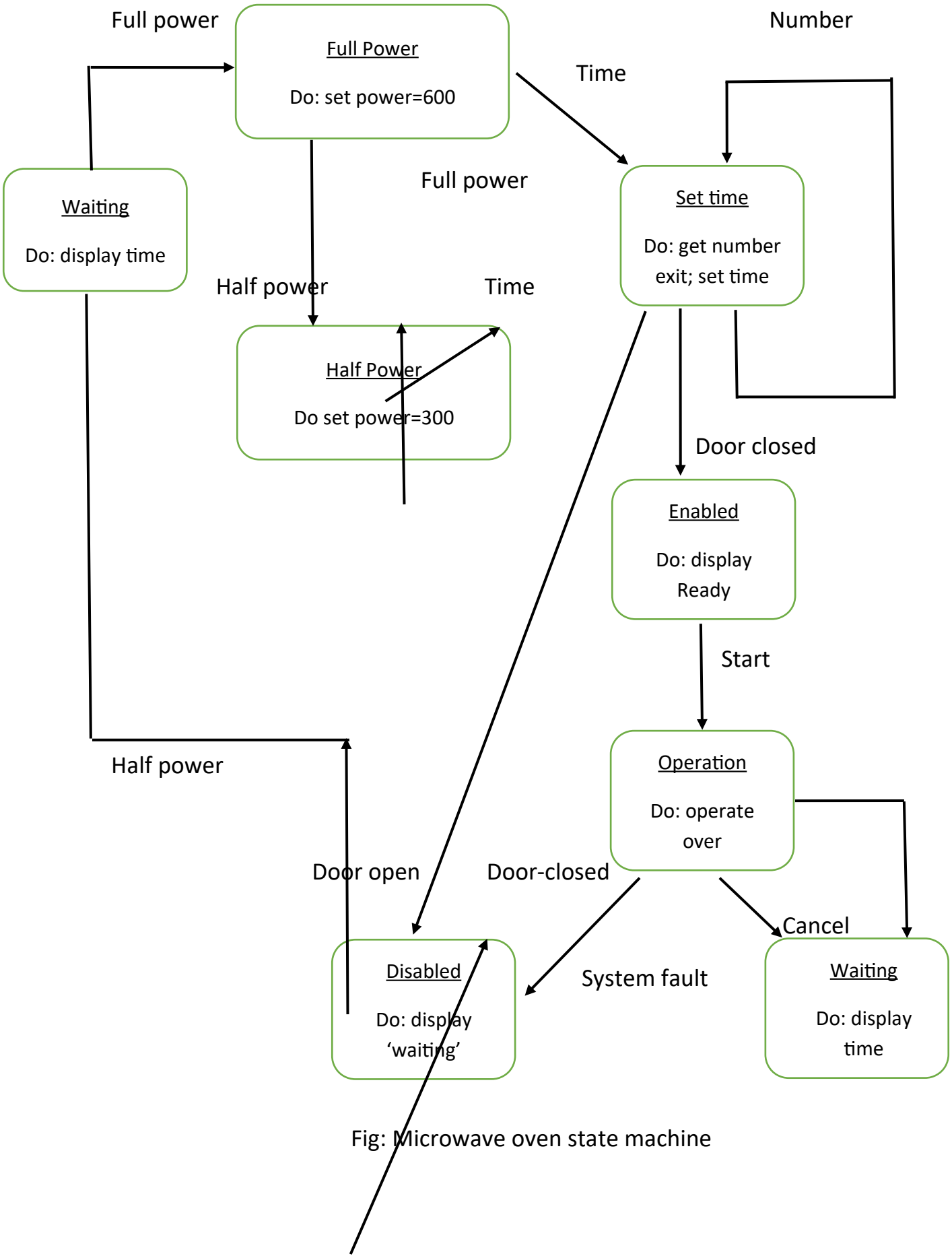

Fig: Microwave oven state machine

State machine modelling

- The effect of stimulus in real-time system may trigger a transition from one state to another.
- Finite state machine can be used for modelling real time system.
- UML includes notations for defining state machine model.

Real time Program

I. Hard teal time system may help to program in assembly language to ensure that deadline are made.
II. Language such as C allows efficient program to be written but do not have constructs to support concurrency or shared resource management.
III. Ada is a language design to support real time system design so includes a general purpose concurrency management.

4.2 Real –time operating system/executives

- Executive is analogous to an OS in a general purpose computer
- Are specialized operating system which manage the processes in RTS.
- Responsible for process management and resource allocation
- Doesn't include facilities such as file management.

Executive components

- Real time clock
  - ✓ Provides information for process scheduling
- Interrupt handle
  - ✓ Manage aperiodic request for service
- Scheduler
  - ✓ Chooses the next process to be run
- Resource manager
  - ✓ Allocate memory and processor resources
- Dispatcher
  - ✓ Start process execution

Non-stop system component

- **Responsible manager**
  - Responsible for dynamic re-configuration of the system software and hardware modules may be replaced and software upgrade stopping the system
- **Fault manager**
  - Responsible for detecting software and hardware faults and taking appropriate actions to ensure that the system continues in operation

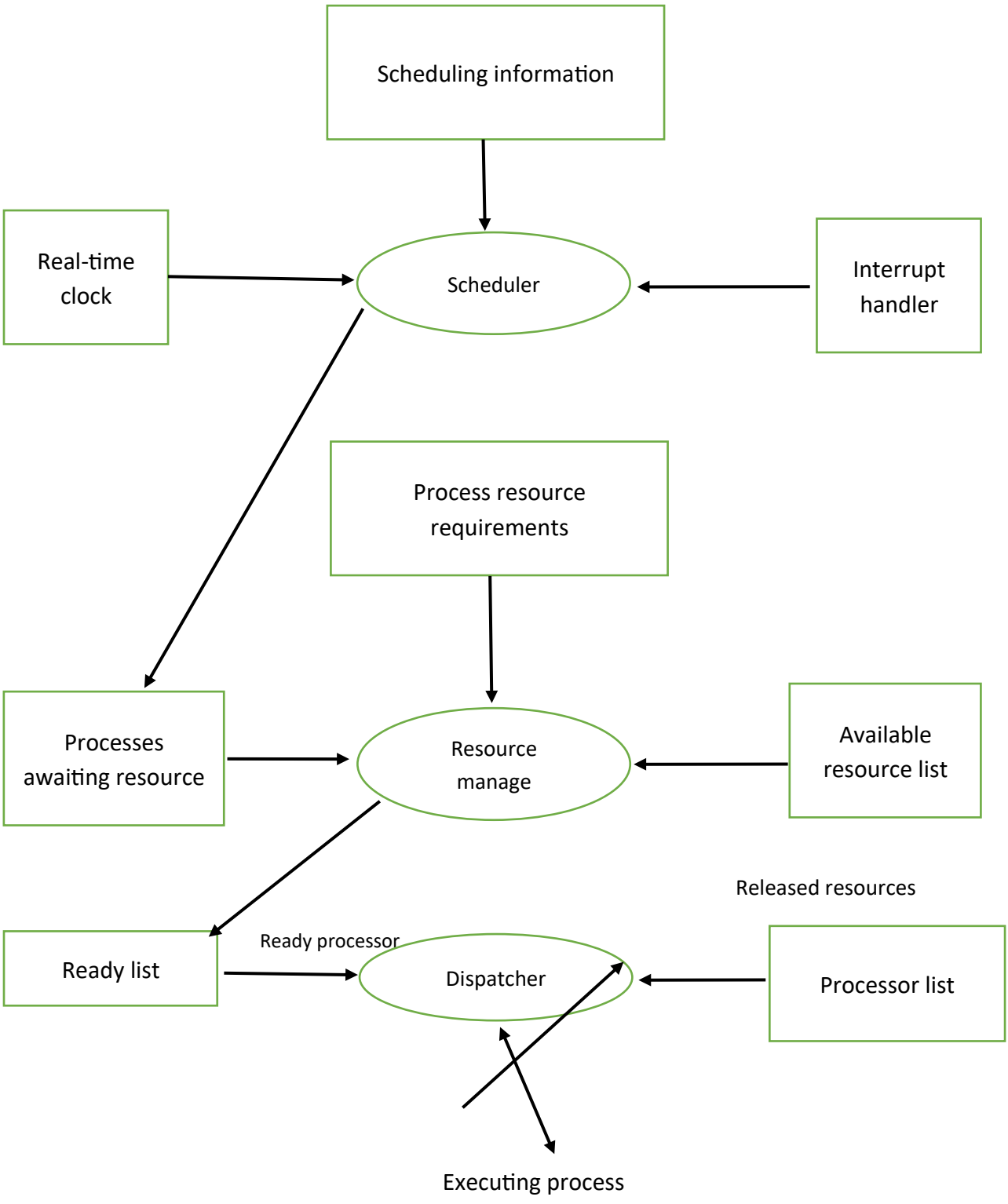

Fig: Real time executive components

4.3 Monitoring & control system

- Important class of real-time systems
- Continuously check sensory & take actions depending on sensor value
- Monitoring system examine sensor & report their results
- Control systems take values & control hardware actuator

Burglar alarm systems

- This system requires to monitor sensors on doors & windows to detect the presence of intrudes in a building.
- When a sensor indicates a break-in, the system switches on light around the area & calls police automatically.
- The system should include provision for operation without a main power supply

Sensor

- Movement detectors, window sensors, door sensors
- 50 window sensor, 50 door sensors, 200 movement sensor
- Voltage drop sensor

Action

- When an intruder is detected, police are called automatically
- Lights are switched on in rooms with active sensors
- An audible alarm is switched on
- The system switches automatically to backup power when a voltage drop is detected

Stimuli to be pressed

- Power failure **->** generated aperiodically by a circuit monitor when received, the system must switched to backup power within 50 ms.
- Introduce alarm **->** stimulus generated by system sensor, response is to call the police, switch on building lights & audible alarm.

Timing constraints

Stimulus/response	**Timing requirement**
1. Power failure	Switch to backup within 50 ms.
2. Door alarm	Polled twice per second
3. Window alarm	" " " "
4. Movement detector	" " " "
5. Audible	Switch on within ½ second
6. Lights switch	" " " " "
7. Communication	Call to police within 2 sec of alarm
8. Voice synthesizer	Message should be available within
.	4 sec of an alarm being raised by a
.	sensor

4.4 Data acquisition system

- Collect data from sensors for subsequent processing & analyzer
- Data collection processes and processing processes may have different periods and deadlines.
- Data collection may be faster than processing
- Circular of ring buffered, are mechanism for smoothing speed differences.

Reactor data collection

- A system collects data from a set of sensor monitoring the neutron flux from a nuclear reactor
- Flux data is placed in a ring buffer for processing. Ring buffer is itself implemented as concurrent processes so that the collection and processing processes may be synchronized.

Reactor flux monitoring

Sensor

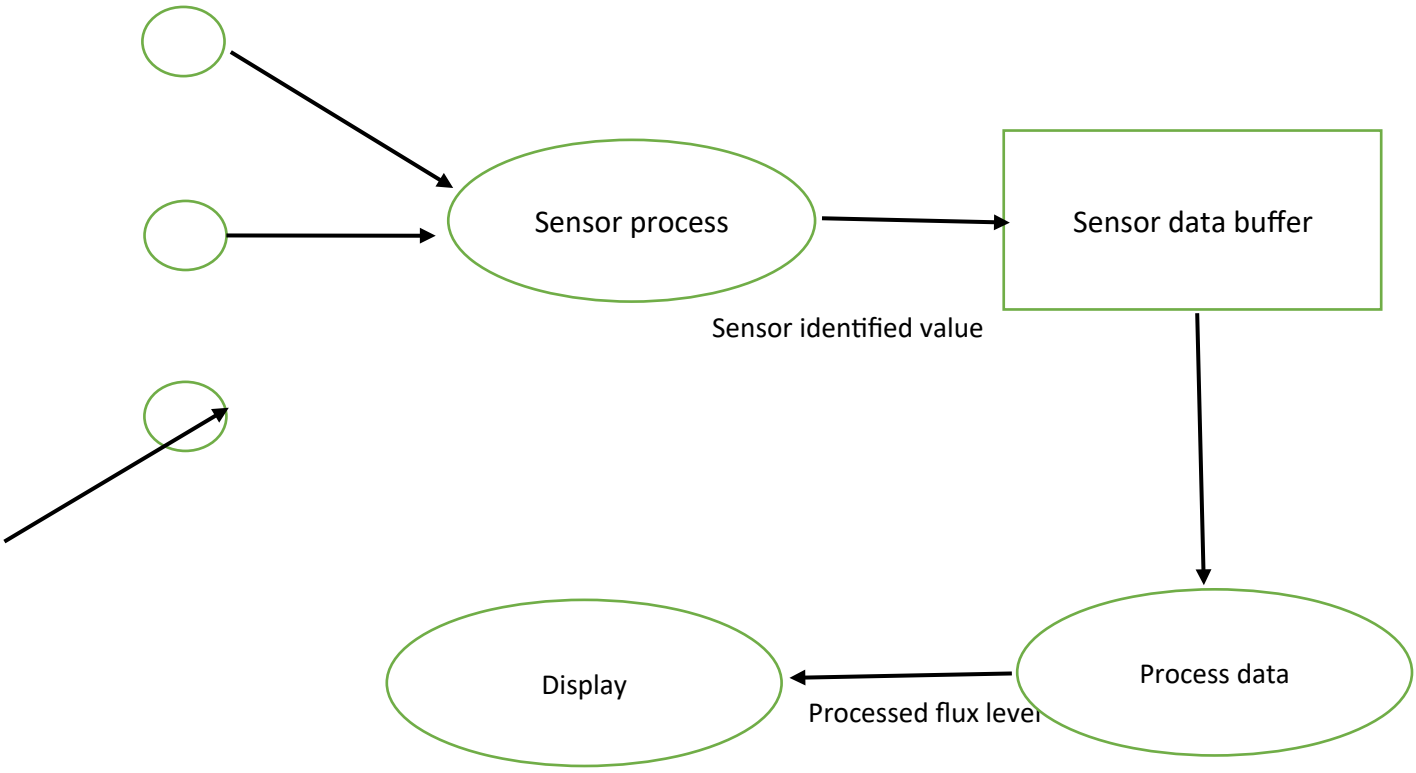

Fig: The architecture of a flux monitoring system

A ring buffer data acquisition

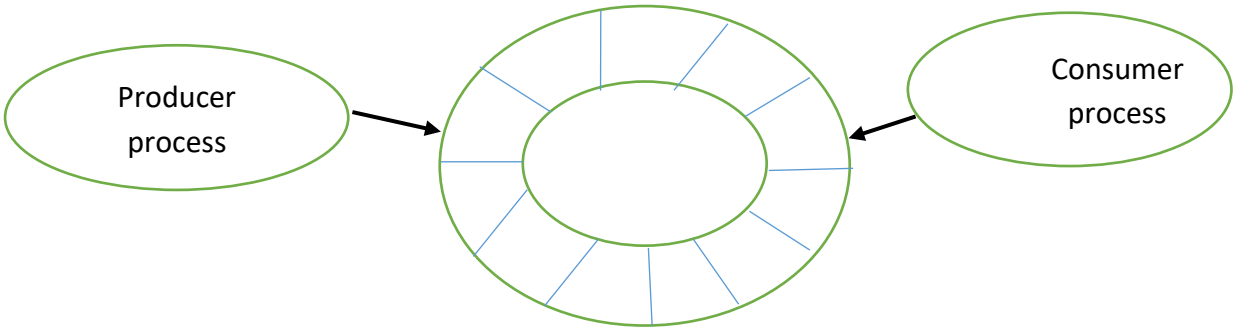

Fig: A ring buffer

- Producer doesn't add while the ring is full
- Consumer doesn't get data while ring buffer is empty