Department of Electronics & Communication Engg.

Course : Real Time Systems (15EC743). **Sem.: 7th (2018-19)**

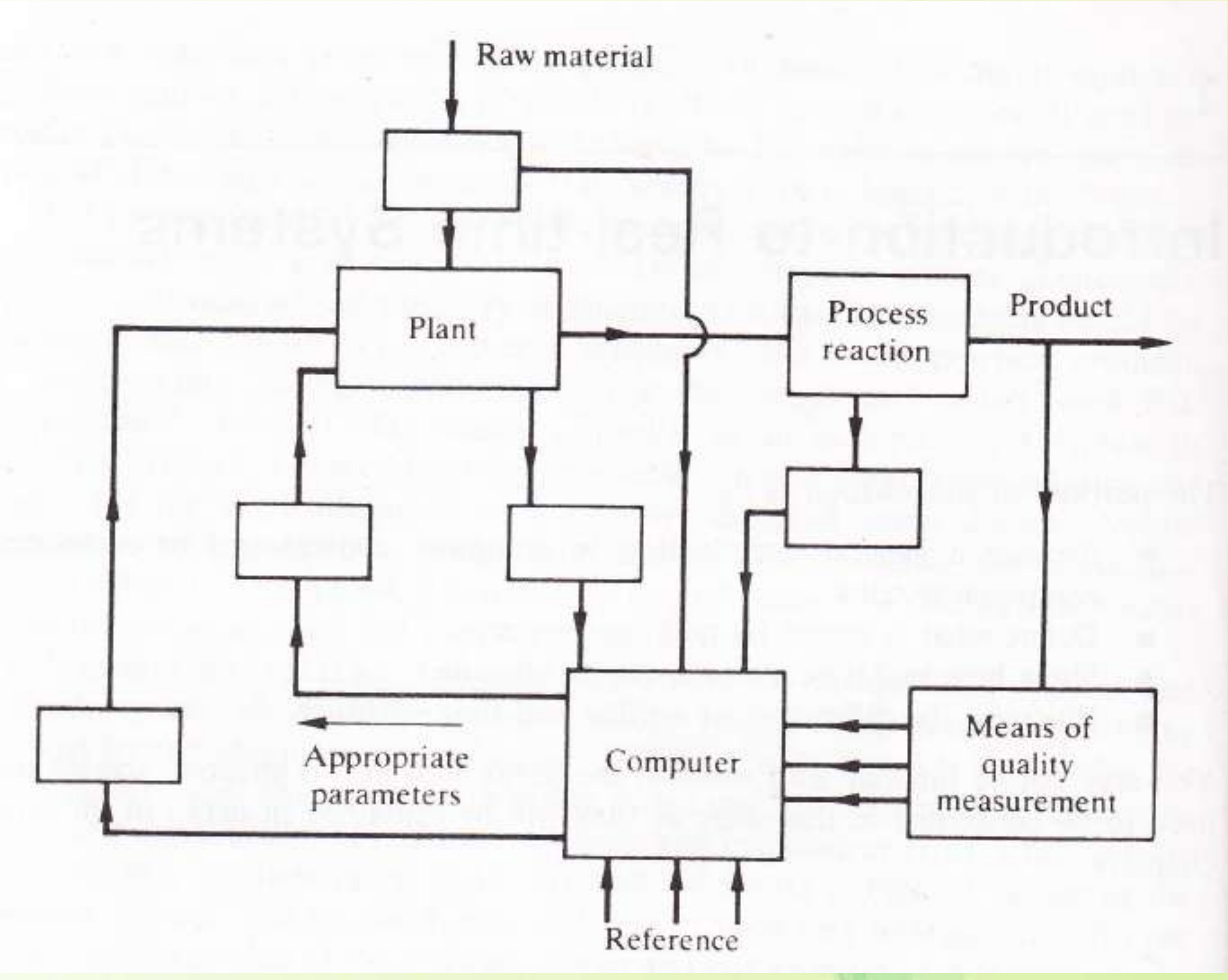
Course Coordinator:

Prof. Sachin S Patil

- ✓ Provide a general introduction to computer control and to embedded computer systems.
- ✓ Define what is meant by real-time systems.
- ✓ Show how real-time systems can be classified.
- ✓ Illustrate the difficulties of writing real-time software.

HISTORICAL BACKGROUND

The earliest proposal to use a computer operating in real time as part of a control system was made in a paper by Brown and Campbell in 1950. The paper contains diagram which shows a computer in both the feedback and feed-forward loops.



During 1957-8 the Monsanto chemical company, in co-operation with the Ramo-wooldridge company, studied the possibility of using computer control and in October 1958 decided to implement a scheme on the ammonia plant at Luling , Louisiana.

The first direct digital control (DDC) computer system was the Ferranti Argus 200 system installed in November 1962 at the ICI ammonia-soda plant at Fleetwood, Lancashire, UK, the planning for which had begun in 1959 (Burkitt, 1965). It was a large system with provision for 120 control loops and 256 measurements, of which 98 and 224 respectively were acrually used on the Fleetwood system

Set points	Loss most undesirable
Valve demand	Presence after controlled stoppage allows computer to gain control of plant immediately and without disturbing the plant (referred to as *bumpless transfer*)
Memory calculations	Loss is tolerable, soon will be updated and only slight disturbance to plant
Future development	Extension to allow for optimisation may require information to be maintained for long periods of time

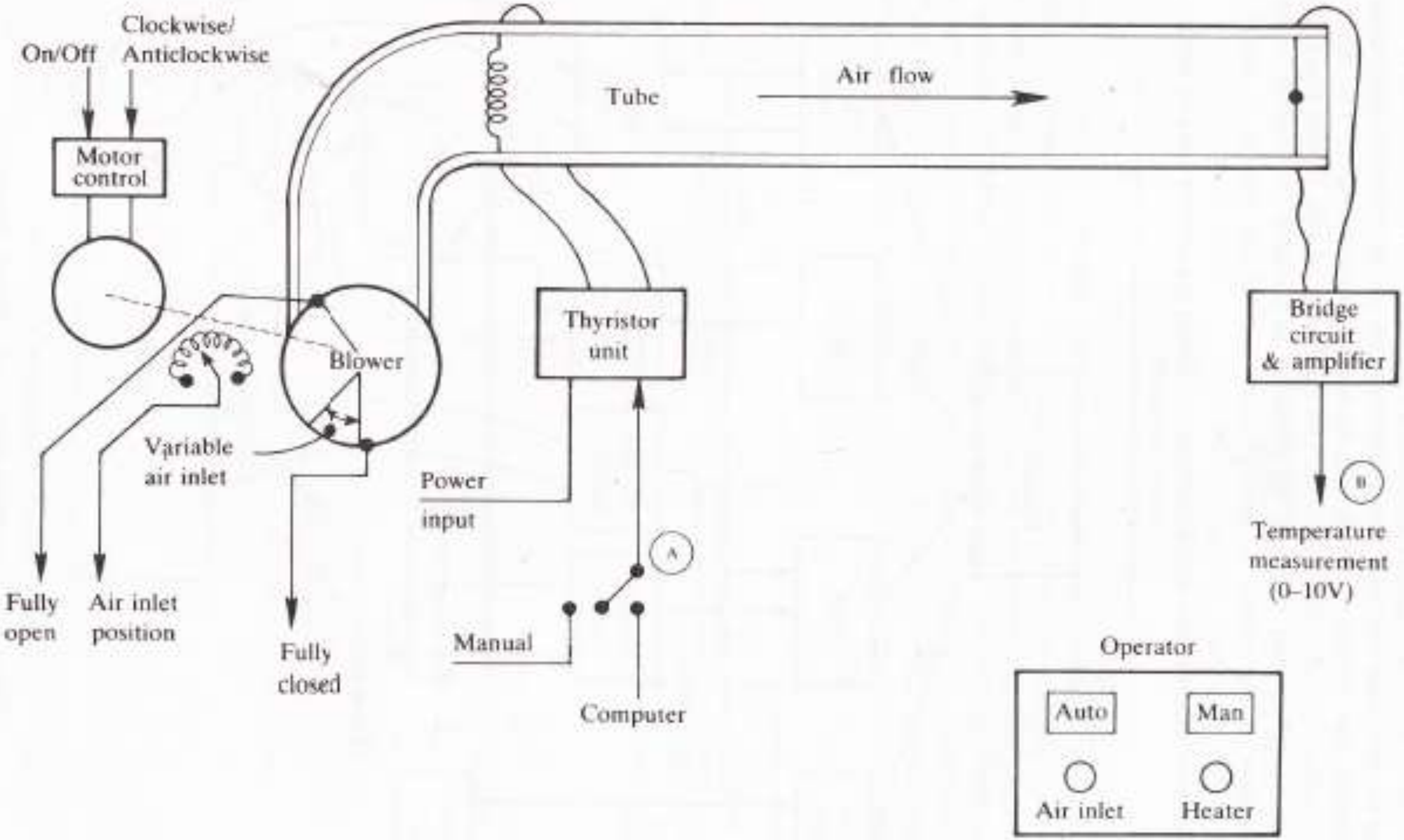


Figure 1.2 A simple plant – a hot-air blower.

As a simple example which illustrates the various operations of a computer control system, let us consider the hot-air blower.

1. A centrifugal fan blows air over a heating element and into a tube.
2. A thermistor bead is placed at the outlet end of the tube and forms one arm of a bridge circuit. The amplified output of the bridge circuit is available at B and provides a voltage, in the range 0 to l0 volts, proportional to temperature.
3. The current supplied to the heating element can be varied by supplying a dc voltage in the range 0 to l0 volts to point A.
4. The position of the air-inlet cover to the fan is adjusted by means of a reversible motor. The motor operates at constant speed and is turned on or off by a logic signal applied to its controller; a second logic signal determines the direction of rotation'
5. A potentiometer wiper is attached to the air-inlet cover and the voltage output is proportional to the position of the cover.
6. Micro-switches are used to detect when the cover is fully open and fully closed.

1. The operator is provided with a panel from which the control system can be switched from automatic to manual control.
2. In manual mode the heat out and fan cover position can be adjusted using potentiometers.
3. Switches are provided to operate the fan and heater. Panel lights indicate fan on, heater on, cover fully open, cover fully closed and auto/manual status. The operator can also adjust the fan cover position.
4. The operation of this simple plant using a computer requires that software be provided to support monitoring, control and actuation of the plant

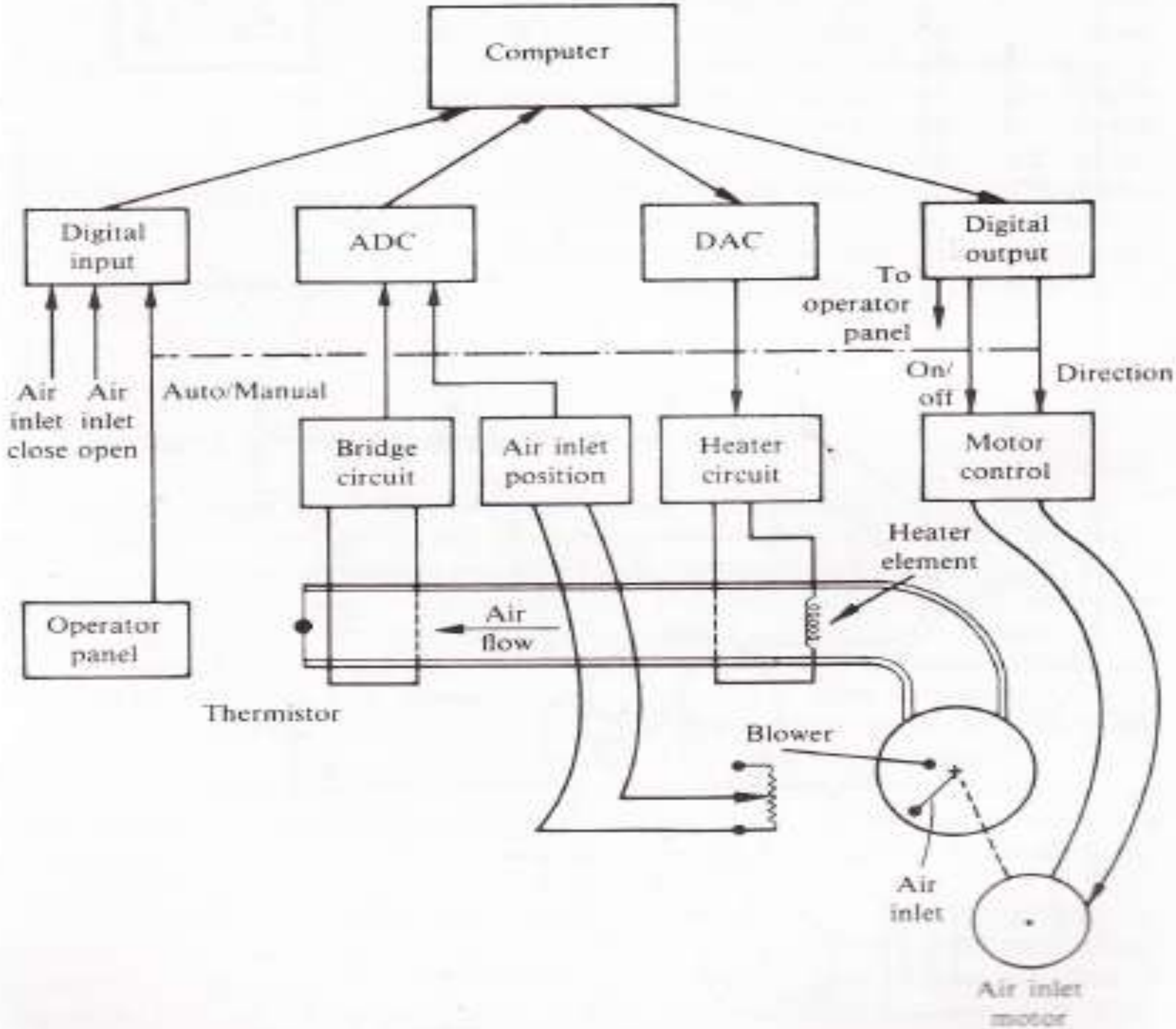


Figure 1.3 Computer control of a hot-air blower.

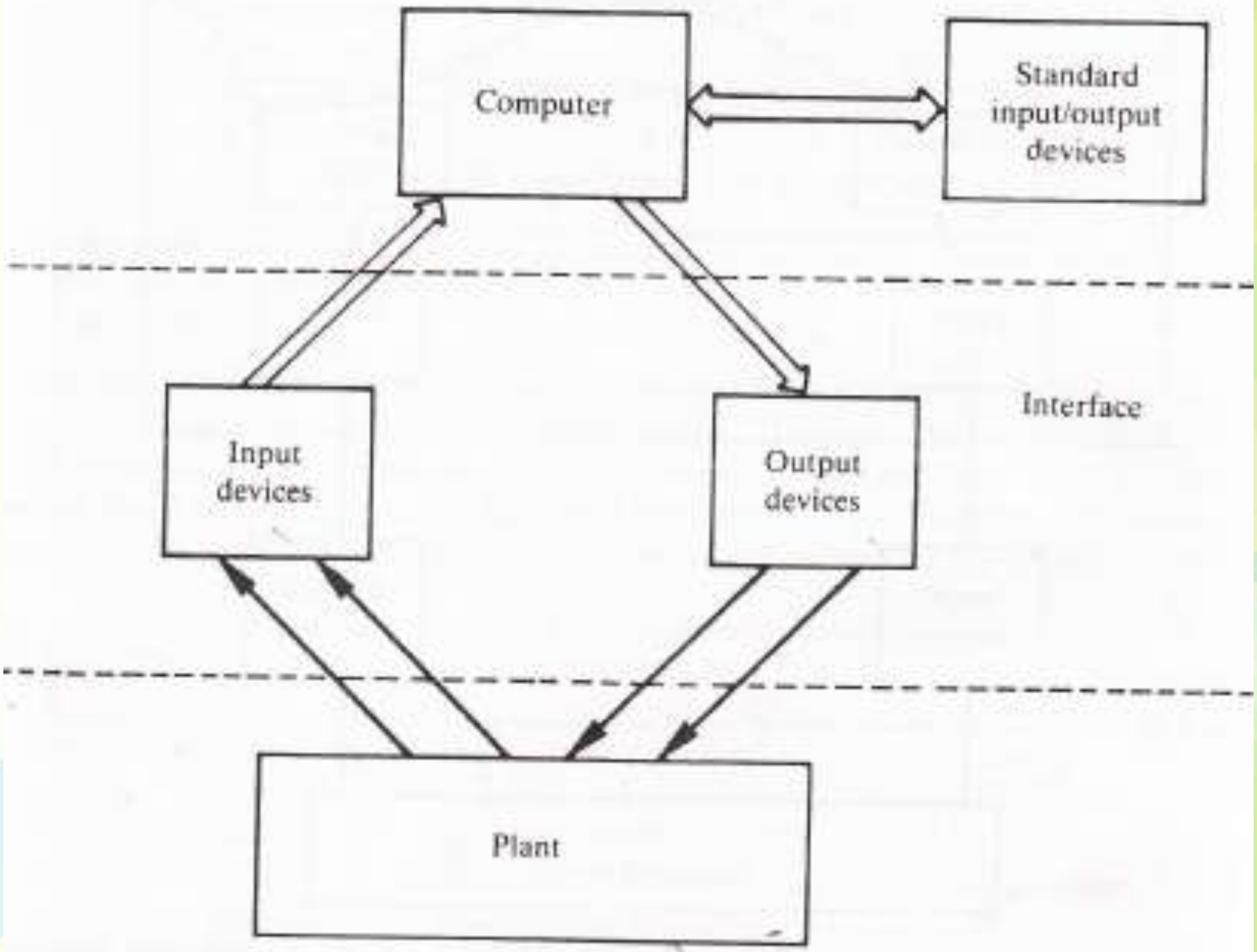


Figure 1.4 Generalised computer system.

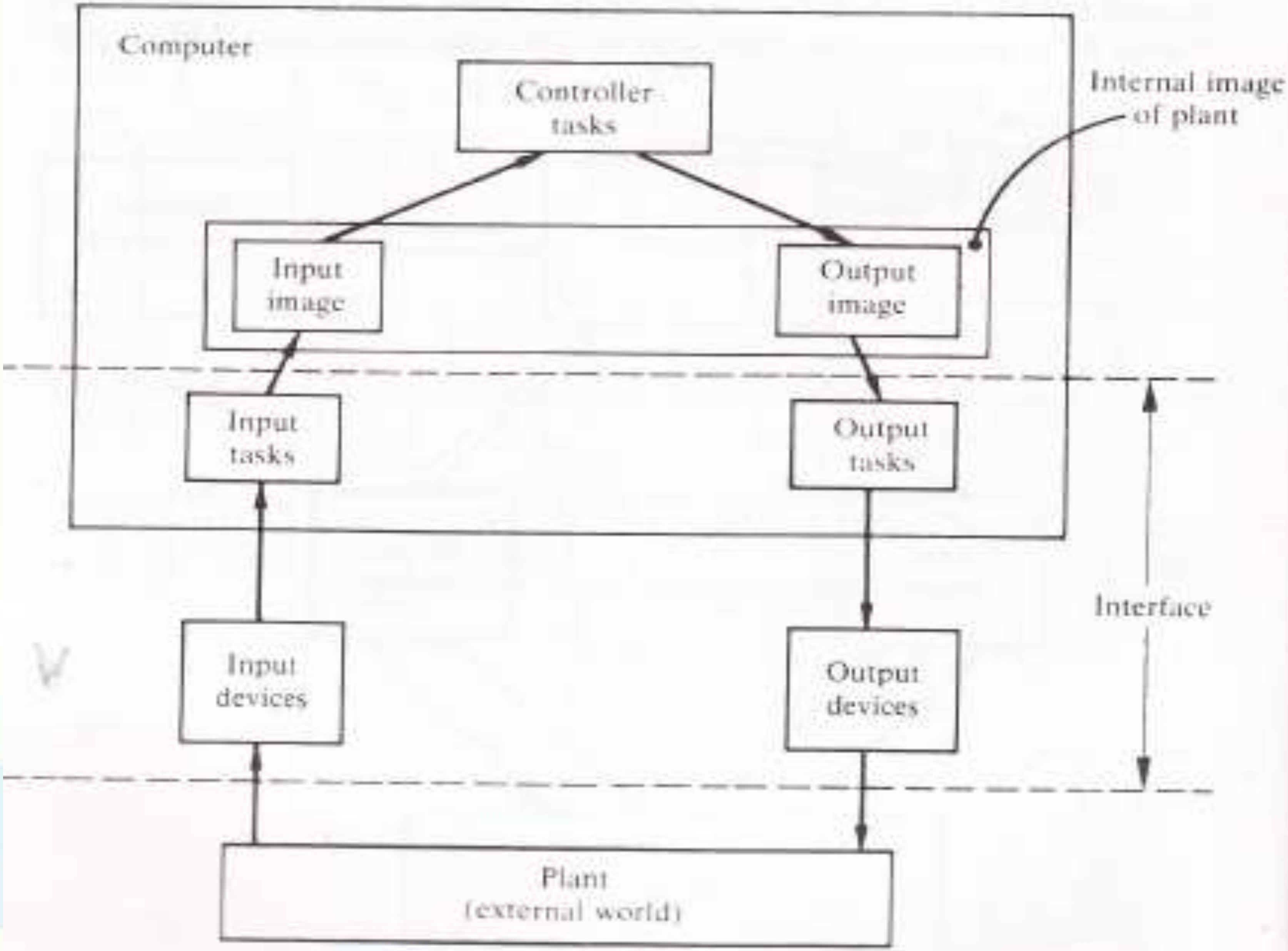


Figure 1.5 Generalised computer control system showing hardware and software interface.

In the simple computer control model we have described above we divided the software tasks to be performed into three major areas:

- plant input tasks;
- plant output tasks; and
- control tasks.

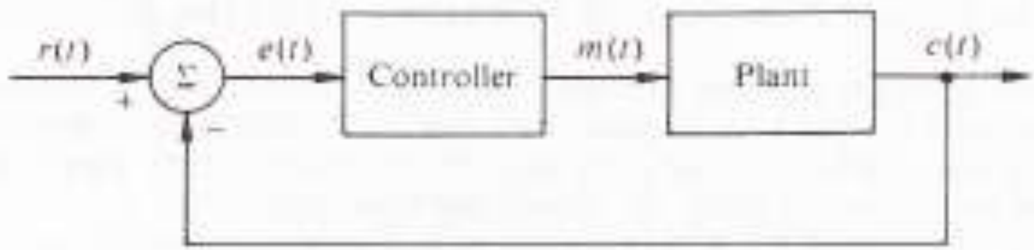


Figure 1.6 Simplified block diagram of continuous feedback control system: $r(t)$ = set point, $c(t)$ = controlled variable, $e(t) = r(t) - c(t)$ = error, and $m(t)$ = manipulated variable.

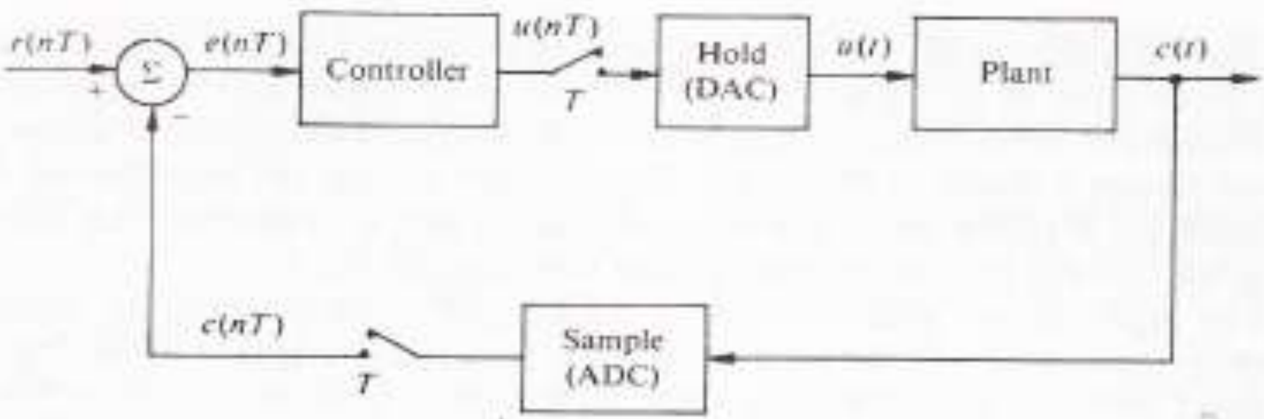


Figure 1.7 Simplified block diagram of a sampled feedback control system: $c(nT)$, $r(nT)$, $e(nT)$, $u(nT)$ are sampled values of $c(t)$, $r(t)$, $e(t)$, $u(t)$ at sample times nT where n is an integer and T is the sampling interval.

REAL-TIME SYSTEMS - DEFINITION

Oxford Dictionary of Computing offers the definition:

Any system. in which the time at which the output is produced is significant. This is usually because the input corresponds to some movement in the physical world, and the output has to relate to that same movement. The lag from input time to output time must be sufficiently small for acceptable timeliness.

we are concerned with the latter type of system and Cooling (1991) offers the definition:

Real-time systems are those which must produce correct responses within a definite time limit. Should computer responses exceed these time bounds then performance degradation and/or malfunction results.

An alternative definition is:

A real-time system, reads inputs from the plant and sends control signals to the plant at times determined by plant operational considerations – not at times limited by the capabilities of the computer system.

We can therefore define a real time program as:

A program for which the correctness of operation depends both on the logical results of the computation and the time at which the results are produced.

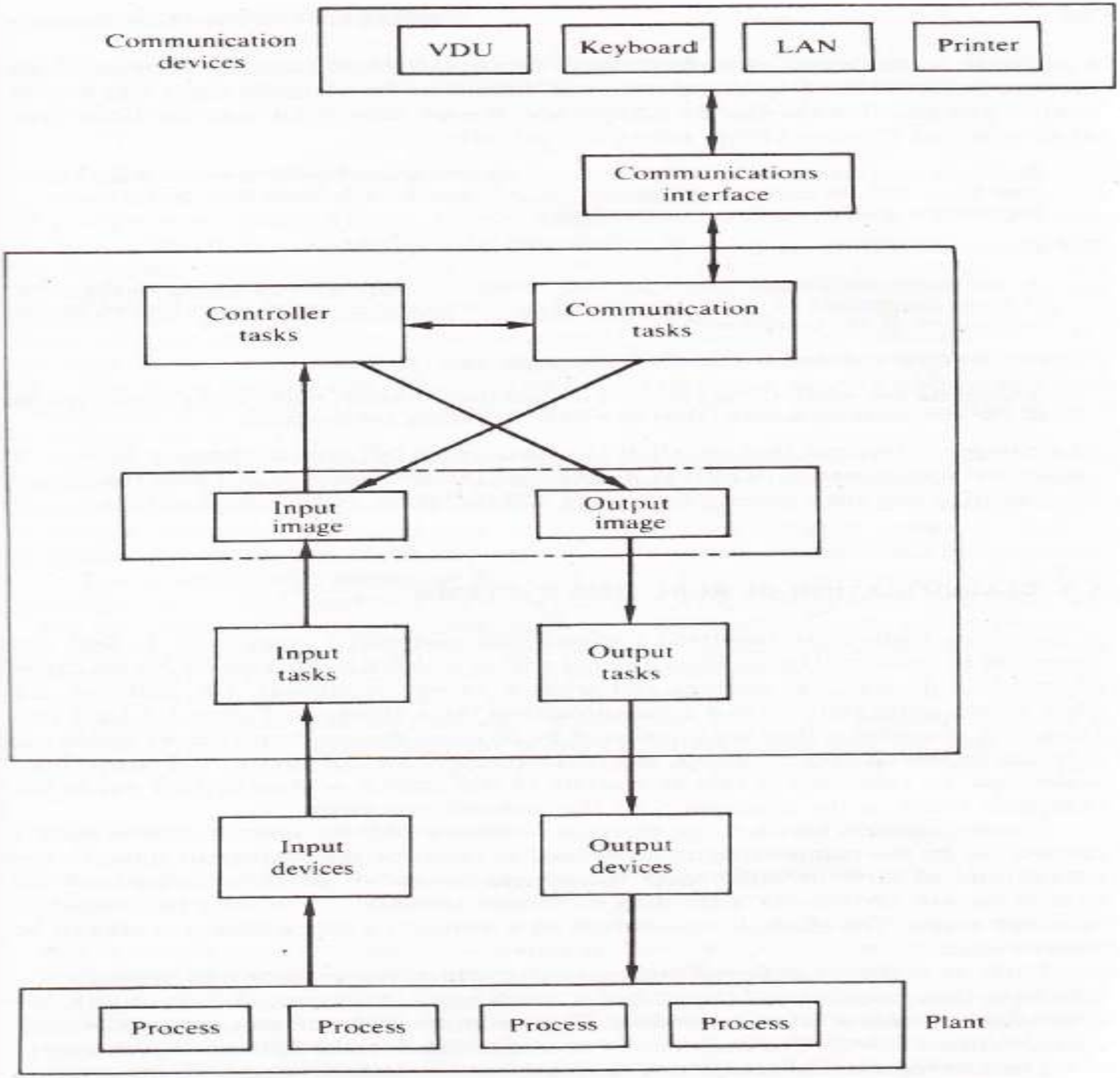


Figure 1.8 Computer control system showing communication tasks.

Classification of Real time systems

A common feature of real-time systems and embedded computers is that the computer is connected to the environment within which it is working by a wide range of interface devices and receives and sends a variety of stimuli

Clock-based Tasks (cyclic, periodic)

Event-based Tasks (aperiodic)

Interactive Systems

Clock-based Tasks (cyclic, periodic)

A process plant operates in real time and thus we talk about the plant time constants.

Time constants may be measured in hours for some chemical processes or in milliseconds for an aircraft system.

For feedback control the required sampling rate will be dependent on the time constant of the process to be controlled.

The shorter the time constant of the process, the faster the required sampling rate. The completion of the operations within the specified time is dependent on the number of operations to be performed and the speed of the computer Synchronisation is usually obtained by adding a clock to the computer system - normally referred to as a real-time clock - and using a signal from this clock to interrupt the operations of the computer at some predetermined fixed time interval.

Clock-based tasks are typically referred to as cyclic or periodic tasks

Event-based Tasks (aperiodic)

There are many systems where actions have to be performed not at particular times Or time intervals but in response to some event.

Typical examples are: turning off a pump or closing a valve when the level of a liquid in a tank reaches a predetermined value.

The specification of event-based systems usually includes a requirement that the system must respond within a given maximum time to a particular event.

Event-based systems normally employ interrupts to inform the computer system that action is required. Some small, simple systems may use polling.

Events occur at non-deterministic intervals and event-based tasks are frequently refered to as aperiodic task

Interactive Systems

Interactive systems probably represent the largest class of real-time systems and cover such systems as automatic bank tellers; reservation systems for hotels, airlines and car rental companies; computerised tills....etc

Superficially this type of system seems similar to the event-based system in that it apparently responds to a signal from the plant. but it is different because it responds at a time determined by the internal state of the computer and without any reference to the environment.

Many interactive systems give the impression that they are clock based in that they are capable of displaying the date and time; they do indeed have a real-time clock which enables them to keep track of time.

'can the system be tightly synchronised to an external process?'
If the answer is 'yes' they are clock based; if it is ,no' then they are interactive.

Time Constraints

It is now common practice to divide real-time systems (and real-time tasks) into two
categories:

1. Hard real-time: these are systems that must satisfy the deadlines on each and every occasion.
2. Soft real-time: these are systems for which an occasional failure to meet a deadline does not comprise the correctness of the system.

A typical example of a hard real-time control system is the temperature control loop of the hot-air blower system.

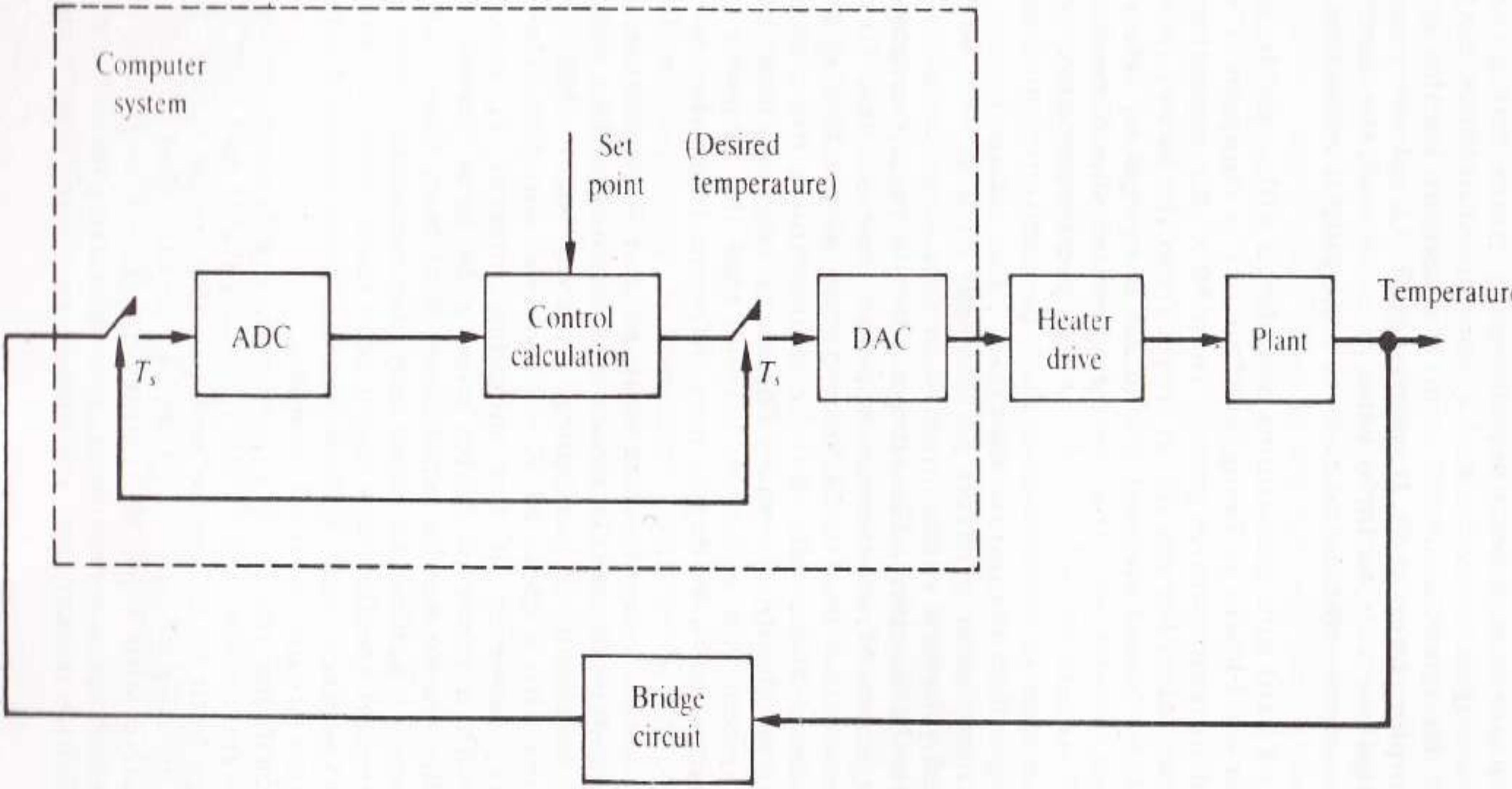


Figure 1.9 Block diagram of computer control system.

We can formally define the constraints as follows:

Hard		*Soft*	
Periodic (cyclic)	*Aperiodic (event)*	*Periodic (cyclic)*	*Aperiodic (event)*
$t_c(i) = t_s \pm a$	$t_e(i) \leqslant T_e$	$\frac{1}{n}\sum_{i=1}^{n} t_c(i) = t_s \pm a$	$\frac{1}{n}\sum_{i=1}^{n} t_e(i) \leqslant T_a$
		$n = T/t_s$	$n = T/t_s$

$t_c(i)$ the interval between the i and $i-1$ cycles,
$t_e(i)$ the response time to the ith occurrence of event e,
t_s the desired periodic (cyclic) interval,
T_e the maximum permitted response time to event e,
T_a the average permitted response time to event e measured over some time interval T,
n the number of occurrences of event e within the time interval T, or the number of cyclic repetitions during the time interval T,
a a small timing tolerance.

Queries …?