

Module

7

Software Engineering  
Issues

# Lesson 35

## Modelling Timing Constraints

# Specific Instructional Objectives

At the end of this lesson, the student would be able to:

- Explain what an event is
- Classify the types of events
- Classify the different types of timing constraints
- Explain what a delay constraint is
- Explain what a deadline constraint is
- Explain what a duration constraint is
- Identify the different types of delay, deadline, and duration constraints associated with a system
- Explain how timing constraints can be modelled
- Explain a Finite State Machine (FSM)
- Explain an Extended Finite State Machine (EFSM)
- Explain how different timing constraints can be modelled using EFSM

## 1. Timing Constraints – An Introduction

The correctness of real-time tasks depend both on the logical correctness of the result, as well as, on the satisfaction of the corresponding timing constraints. The timing constraints as we shall see in this section, in fact, apply to certain events in a system. These events may be generated by the tasks themselves or the environment of the system. An example of such an event is the event of activation of a motor. Remember that the results may be generated at different times and it may not be in the form of a single one-time result. We must first properly characterize the events in a system, to understand the timing behavior of real-time systems.

### 1.1. Events in a System

An event may be generated either by the system or its environment. Based on this consideration, events can be classified into the following two types:

**Stimulus Events:** Stimulus events are generated by the environment and act on the system. These events can be produced asynchronously (i.e. aperiodically). For example, a user pressing a button on a telephone set generates a stimulus event to act on the telephone system. Stimulus events can also be generated periodically. As an instance, consider the periodic sensing of the temperature of the reactor in a nuclear plant.

**Response Events:** Response events are usually produced by the system in response to some stimulus events. Response events act on the environment. For example, consider a chemical plant where as soon as the temperature exceeds 100° C, the system responds by switching off the heater. Here, the event of temperature exceeding 100° C is the stimulus and switching off of the heater is the response. Response events can either be periodic or aperiodic.

An event may either be instantaneous or may have certain duration. For example, a button press event is described by the duration for which the button was kept pressed. Some authors argue that durational events are really not a basic type of event, but can be expressed using other events. In fact, it is possible to consider a duration event as a combination of two events: a start event and an end event. For example, the button press event can be described by a combination of ‘start button press’ and ‘end button press’ events. However, it is often convenient to retain the notion of a durational event. In this text, we consider durational events as a special class of events. Using the preliminary notions about events discussed in this subsection, we classify various types of timing constraints in subsection 1.7.1.

## 1.2. Classification of Timing Constraints

A classification of the different types of timing constraints is important. Not only would it give us an insight into the different types of timing constraints that can exist in a system, but it can also help us to quickly identify the different timing constraints that can exist from a casual examination of a problem. That is, in addition to better understanding of the behavior of a system, it can also let us work out the specification of a real-time system accurately.

Different timing constraints associated with a real-time system can broadly be classified into performance and behavioral constraints.

Performance constraints are the constraints that are imposed on the response of the system. Behavioral constraints are the constraints that are imposed on the stimuli generated by the environment.

Behavioral constraints ensure that the environment of a system is well behaved, whereas performance constraints ensure that the computer system performs satisfactorily.

Each of *performance* and *behavioral constraints* can further be classified into the following three types:

- Delay Constraint
- Deadline Constraint
- Duration Constraint

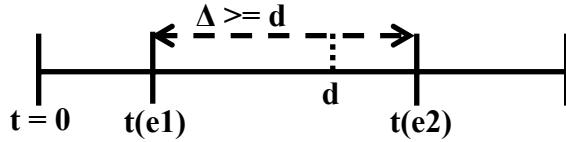
These three classes of constraints are explained in the subsequent sections.

### 1.2.1. Delay Constraints

A delay constraint captures the minimum time (delay) that must elapse between the occurrence of two arbitrary events  $e_1$  and  $e_2$ . After  $e_1$  occurs, if  $e_2$  occurs earlier than the minimum delay, then a delay violation is said to occur. A delay constraint on the event  $e_2$  can be expressed more formally as follows:

$$t(e_2) - t(e_1) \geq d$$

where  $t(e_2)$  and  $t(e_1)$  are the time stamps on the events  $e_2$  and  $e_1$  respectively and  $d$  is the minimum delay specified from  $e_2$ . A delay constraint on the events  $e_2$  with respect to the event  $e_1$  is shown pictorially in Fig. 35.1. In Fig. 35.1s,  $\Delta$  denotes the actual separation in time between the occurrence of the two events  $e_1$  and  $e_2$  and  $d$  is the required minimum separation between the two events (delay). It is easy to see that  $e_2$  must occur after at least  $d$  time units have elapsed since the occurrence of  $e_1$ ; otherwise we shall have a delay violation.

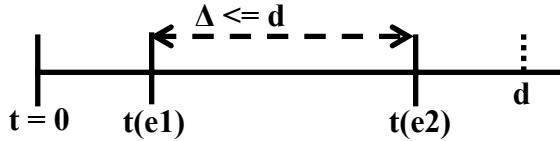


**Fig. 35.1 Delay Constraint between two events  $e_1$  and  $e_2$**

### 1.2.2. Deadline Constraints

A deadline constraint captures the permissible maximum separation between any two arbitrary events  $e_1$  and  $e_2$ . In other words, the second event (i.e.  $e_2$ ) must follow the first event (i.e.  $e_1$ ) within the permissible maximum separation time. Consider that  $t(e_1)$  and  $t(e_2)$  are the time stamps on the occurrence of the events  $e_1$  and  $e_2$  respectively and  $d$  is the deadline as shown in Fig. 35.2. In Fig. 35.2,  $\Delta$  denotes the actual separation between the time of occurrence of the two events  $e_1$  and  $e_2$ , and  $d$  is the deadline. A deadline constraint implies that  $e_2$  must occur within  $d$  time units of  $e_1$ 's occurrence. We can alternatively state that  $t(e_1)$  and  $t(e_2)$  must satisfy the constraint:

$$t(e_2) - t(e_1) \leq d$$



**Fig. 35.2 Deadline Constraint between two events  $e_1$  and  $e_2$**

The deadline and delay constraints can further be classified into two types each based on whether the constraint is imposed on the stimulus or on the response event. This has been explained with some examples in section 1.3.

### 1.2.3. Duration Constraints

A duration constraint on an event specifies the time period over which the event acts. A duration constraint can either be minimum type or maximum type. The minimum type duration constraint requires that once the event starts, the event must not end before a certain minimum duration; whereas a maximum type duration constraint requires that once the event starts, the event must end before a certain maximum duration elapses.

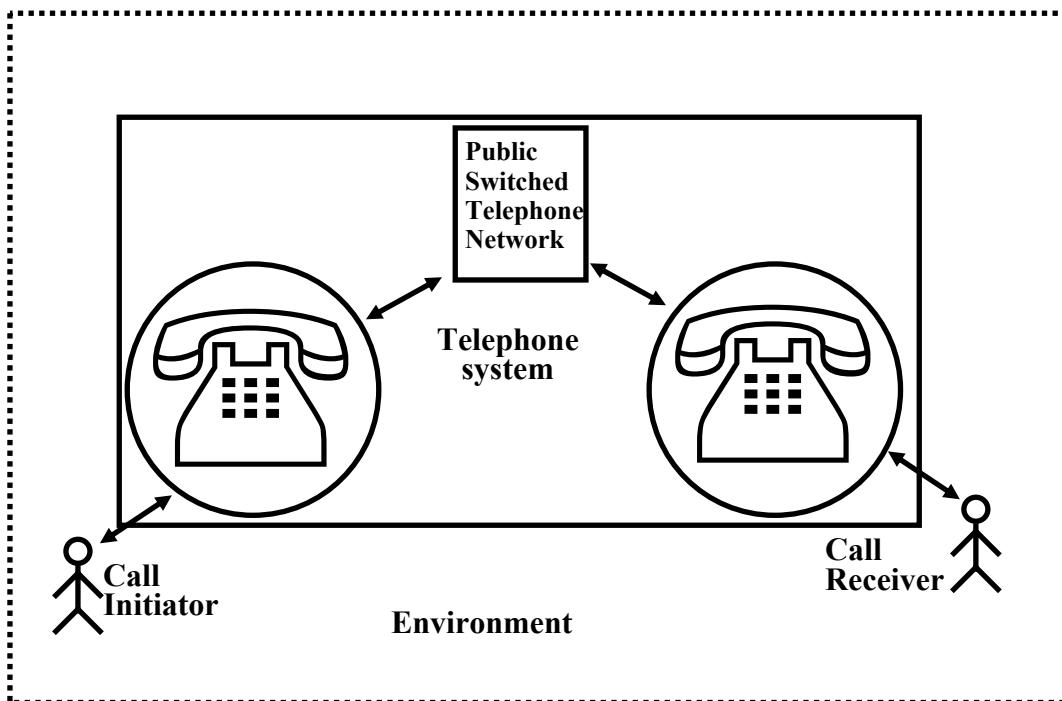


Fig. 35.3 Schematic Representation of a Telephone System

### 1.3. Examples of Different Types of Timing Constraints

We illustrate the different classes of timing constraints by using the examples from a telephone system discussed in. A schematic diagram of a telephone system is given in Fig. 35.3. Note that I have intentionally drawn an old styled telephone, because its operation is easier to understand! Here, the telephone handset and the Public Switched Telephone Network (PSTN) are considered as constituting the computer system and the users as forming the environment. In the following, we give a few simple example operations of the telephone system to illustrate the different types of timing constraints.

**Deadline constraints:** In the following, we discuss four different types of deadline constraints that may be identified in a real-time system depending on whether the two events involved in a deadline constraint are stimulus type or response type.

**Stimulus–Stimulus (SS):** In this case, the deadline is defined between two stimuli. This is a behavioral constraint, since the constraint is imposed on the second event which is a stimulus. An example of an SS type of deadline constraint is the following:

*Once a user completes dialling a digit, he must dial the next digit within the next 5 seconds; otherwise an idle tone is produced.*

In this example, the dialing two consecutive digits represent the two stimuli to the telephone system.

**Stimulus–Response (SR):** In this case, the deadline is defined on the response event, measured from the occurrence of the corresponding stimulus event. This is a performance constraint, since the constraint is imposed on a response event. An example of an SR type of deadline constraint is the following:

Once the receiver of the hand set is lifted, the dial tone must be produced by the system within 2 seconds, otherwise a beeping sound is produced until the handset is replaced. In this example, the lifting of the receiver hand set represents a stimulus to the telephone system and production of the dial tone is the response.

**Response-Stimulus (RS):** Here the deadline is on the production of response counted from the corresponding stimulus. This is a behavioral constraint, since the constraint is imposed on the stimulus event. An example of an RS type of deadline constraint is the following:

*Once the dial tone appears, the first digit must be dialed within 30 seconds, otherwise the system enters an idle state and an idle tone is produced.*

**Response-Response (RR):** An RR type of deadline constraint is defined on two response events. In this case, once the first response event occurs, the second response event must occur before a certain deadline. This is a performance constraint, since the timing constraint has been defined on a response event. An example of an RR type of deadline constraint is the following:

*Once the ring tone is given to the callee, the corresponding ring back tone must be given to the caller within two seconds, otherwise the call is terminated.*

Here ring back tone and the corresponding ring tone are the two response events.

**Delay Constraints:** We can identify only one type of delay constraint (SS type) in the telephone system example that we are considering. However, in other problems it may be possible to identify different types of delay constraints. An SS type of a delay constraint is a behavioral constraint. An example of an SS type of delay constraint is the following:

*Once a digit is dialled, the next digit should be dialled after at least 1 second. Otherwise, a beeping sound is produced until the call initiator replaces the handset.*

Here the delay constraint is defined on the event of dialling of the next digit (stimulus) after a digit is dialled (also a stimulus).

**Duration Constraint:** A duration constraint on an event specifies the time interval over which the event acts. An example of a duration constraint is the following:

*If you press the button of the handset for less than 15 seconds, it connects to the local operator. If you press the button for any duration lasting between 15 to 30 seconds, it connects to the international operator. If you keep the button pressed for more than 30 seconds, then on releasing it would produce the dial tone.*

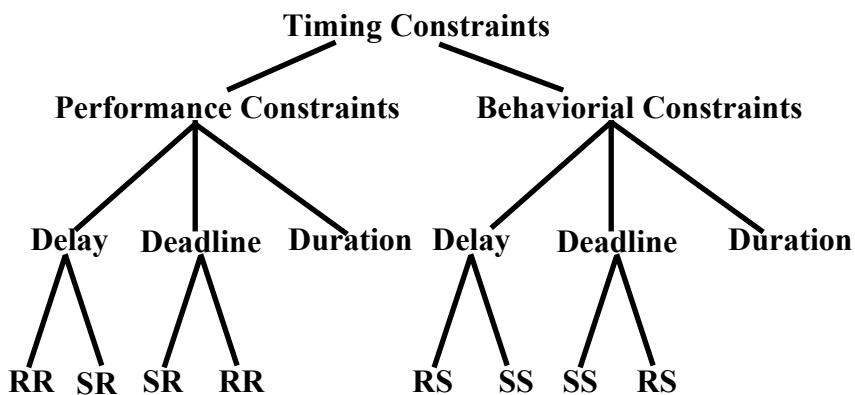


Fig. 35.4 Classification of Timing Constraints

A classification of the different types of timing constraints that we discussed in this section is shown in Fig. 35.4. Note that a performance constraint can either be delay, deadline, or durational type. The delay or deadline constraints on performance can either be RR or RS type. Similarly, the behavioral constraints can either be delay, deadline, or durational type. The delay or deadline constraints on behavior of environment can either be RS or SS type.

## 2. Modelling Timing Constraints

In this section, we describe how the timing constraints identified in Sec. 1.2 can be modelled. Modelling time constraints is very important since once a model of the time constraints in a system is constructed, it can serve as a formal specification of the system. Further, if all the timing constraints in a system are modelled accurately, then it may even be used to automatically generate code from it. Besides serving as a specification, modelling time constraints can help to verify and understand a real-time system.

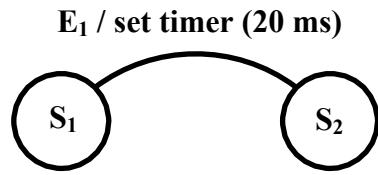
### 2.1. The Finite State Machine (FSM)

The modelling approach we discuss here is based on *Finite State Machines* (FSMs). An FSM is a powerful tool which has long been used to model traditional systems. In an FSM, a state is defined in terms of the values assumed by some attributes. For example, the states of an elevator may be denoted in terms of its directions of motion. Here direction is the attribute, based on which the states up, down, and stationery are defined.

In an FSM model, at any point of time a system can be in any one of a (possibly infinite) number of states. A state is represented by a circle. The system changes state due to events that change the values of, or relations among the state variables. A state change is also called a state transition. A transition causing event may either be an interface event that are transmitted between the environment and the computer system or it could also be an internal event that is generated and consumed solely within the system. A transition from one state to another is represented by drawing a directed arc from the source to the destination (see Fig.35.5). The event causing a transition is annotated on the arc. We keep our discussions of FSM to the bare minimum since we assume that the reader is familiar with basic FSM modelling of traditional systems.

### 2.2. Extended Finite State Machine (EFSM)

We use an Extended Finite State Machine to model time constraints. EFSM extends the traditional FSM by incorporating the action of setting a timer and the expiry event of a timer. The notations we use for construction of EFSMs are simple and straightforward. Therefore rather than introducing them formally, we have illustrated them through an example in Fig. 35.5. The example shown in Fig. 35.5 describes that if an event e1 occurs when the current state of the system is s1, then an action will be taken by setting a timer to expire in the next 20 milliseconds and the system transits to state s2.

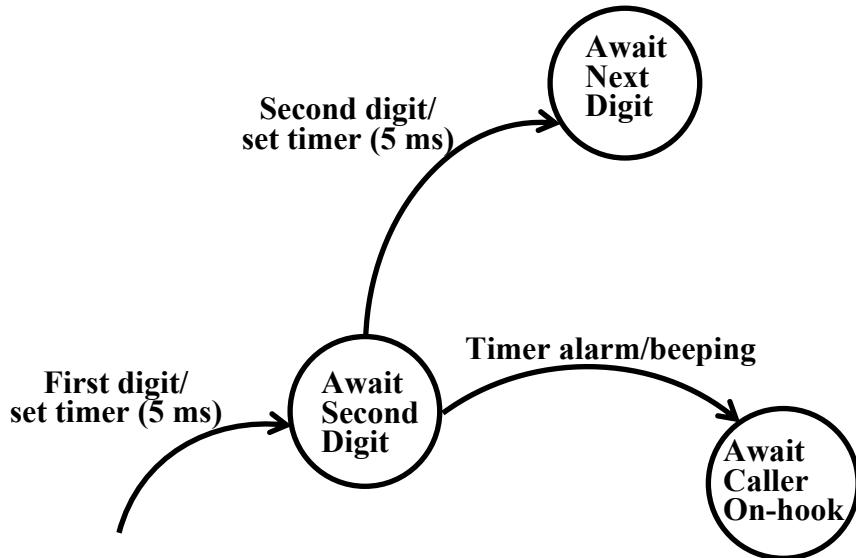


**Fig. 35.5 Conventions Used in Drawing an EFSM**

We have already discussed that events can be considered to be of two types: stimulus events and response events. We had also discussed different types of timing constraints in Section 1.3. Now we explain how these constraints can be modelled by using EFSMs.

### 2.2.1. Stimulus-Stimulus (SS)

Let us consider the example of an SS type of deadline constraint we had discussed in Section 1.3: *Once the first digit has been dialled on the telephone handset, the next digit must be dialled within the next 5 milliseconds*. This has been modelled in Fig. 35.6. In Fig. 35.6, we can observe that as soon as the first digit is dialled, the system enters the “*Await Second Digit*” state and the timer is set to 20 milliseconds. If the next digit does not appear within 20 milliseconds, then the timer alarm expires and the system enters the “*Await Caller On-hook*” state and a beeping sound is produced. If the second digit occurs before 20 milliseconds, then the system transits to the “*Await Next Digit*” state.

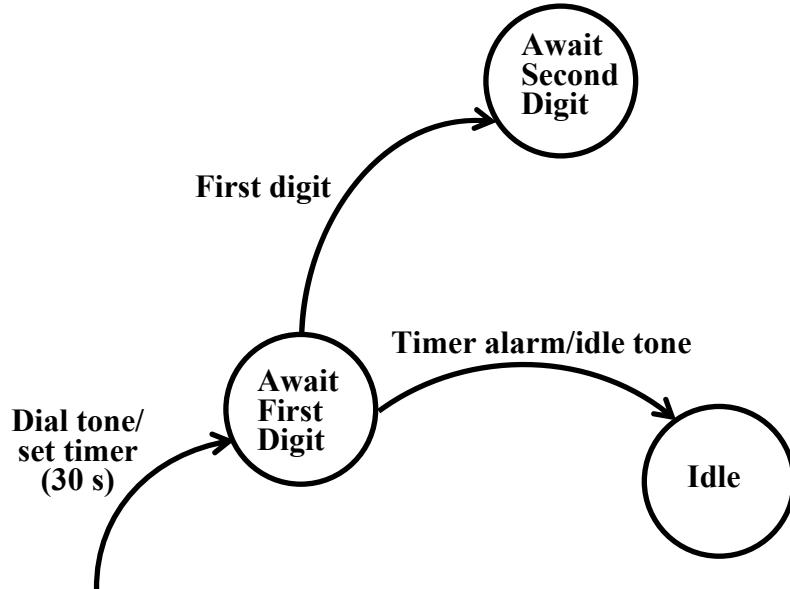


**Fig. 35.6 Model of an SS Type of Deadline Constraint**

### 2.2.2. Response-Stimulus (RS)

In Sec. 1.3, we had considered the following example of an RS type of deadline constraint: *Once the dial tone appears, the first digit must be dialed within 30 seconds, otherwise the system enters an idle state and an idle tone is produced*.

The EFSM model for this constraint is shown in Fig. 35.7. In Fig. 35.7, as soon as dial tone appears, a timer is set to expire in 30 seconds and the system transits to the “*Await First Digit*” state. If the timer expires before the first digit arrives, then the system transits to an idle state where an idle tone is produced. Otherwise, if the digit appears first, then the system transits to the “*Await Second Digit*” state.

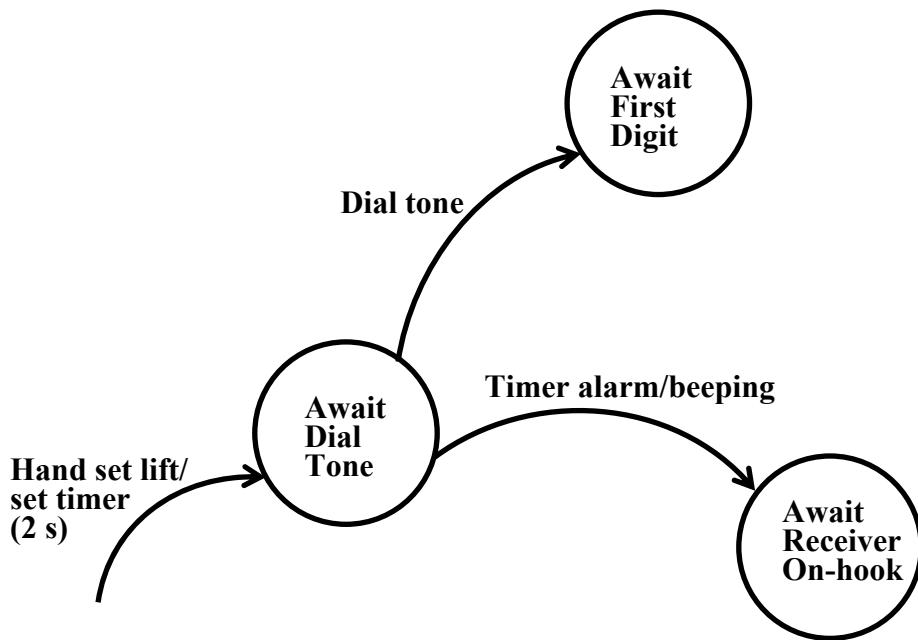


**Fig. 35.7 Model of an RS Type of Deadline**

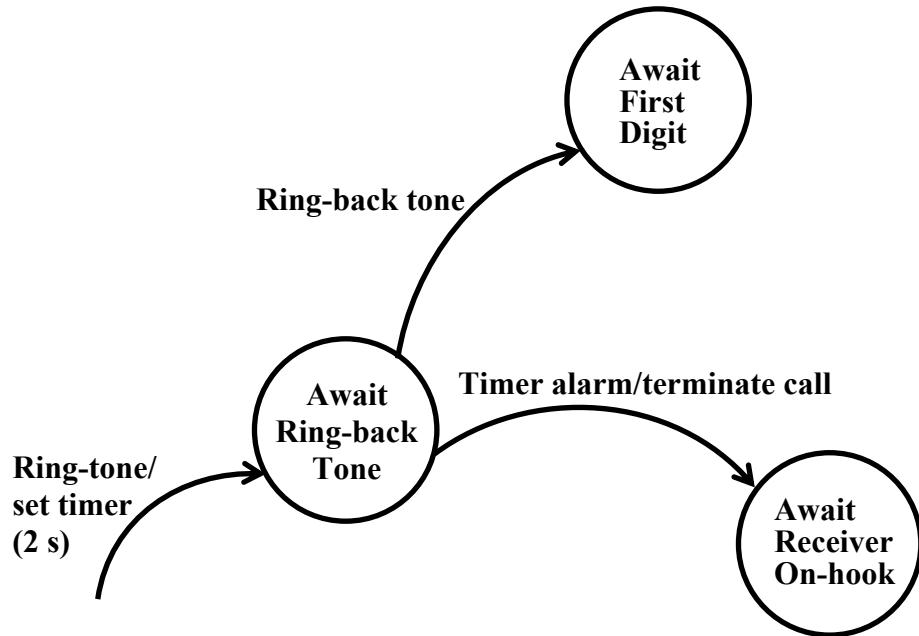
### 2.2.3. Stimulus–Response (SR)

In Sec. 1.3, we had considered the following example of an SR type of deadline constraint: *Once the receiver of the hand set is lifted, the dial tone must be produced by the system within 20 seconds, otherwise a beeping sound is produced until the handset is replaced.*

The EFSM model for this constraint is shown in Fig. 35.8. As soon as the handset is lifted, a timer is set to expire after 2 sec and the system transits to “*Await Dial Tone*” state. If the dial tone appears first, then the system transits to “*Await First Digit*” state. Otherwise, it transits to “*Await Receiver On-hook*” state.



**Fig. 35.8 Model of an SR Type of Deadline**



**Fig. 35.9 Model of an RR Type of Deadline Constraint**

## 2.2.4. Response–Response (RR)

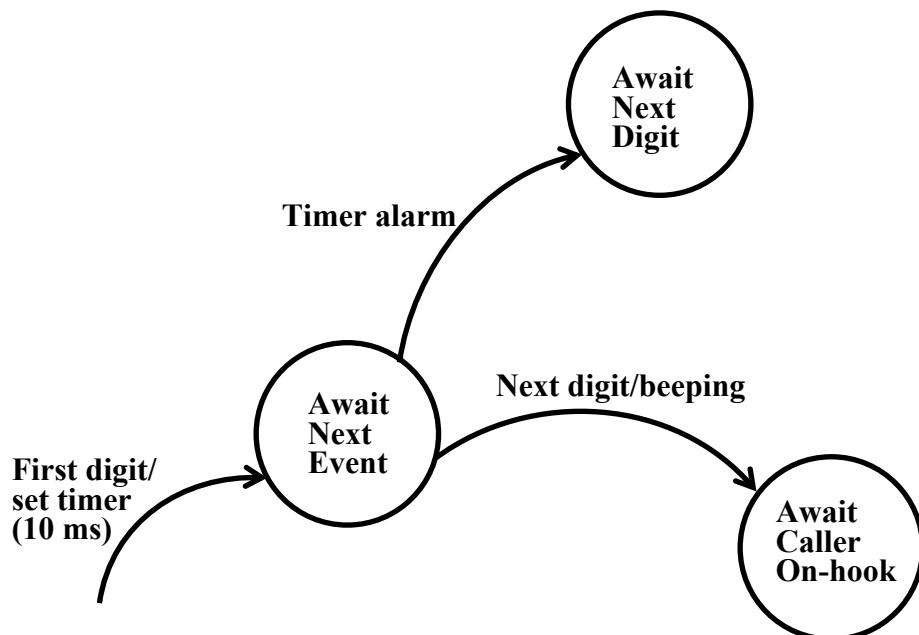
In Sec. 1.3, we had considered the following example of an RR type of constraint: *Once the ring tone is given to the callee, the corresponding ring back tone must be given to the caller within two seconds, otherwise the call is terminated.*

The EFSM model for this constraint is shown in Fig. 35.9. In Fig. 35.9, as soon as the ring tone is produced, the system transits to “Await Ring-back Tone” state, and a timer is set to expire

in 2 seconds. If the ring-back tone appears first, the system transits to “*Await First Digit*” state, else it enters “*Await Receiver On-hook*” state, and the call is terminated.

### 2.2.5. Delay Constraint

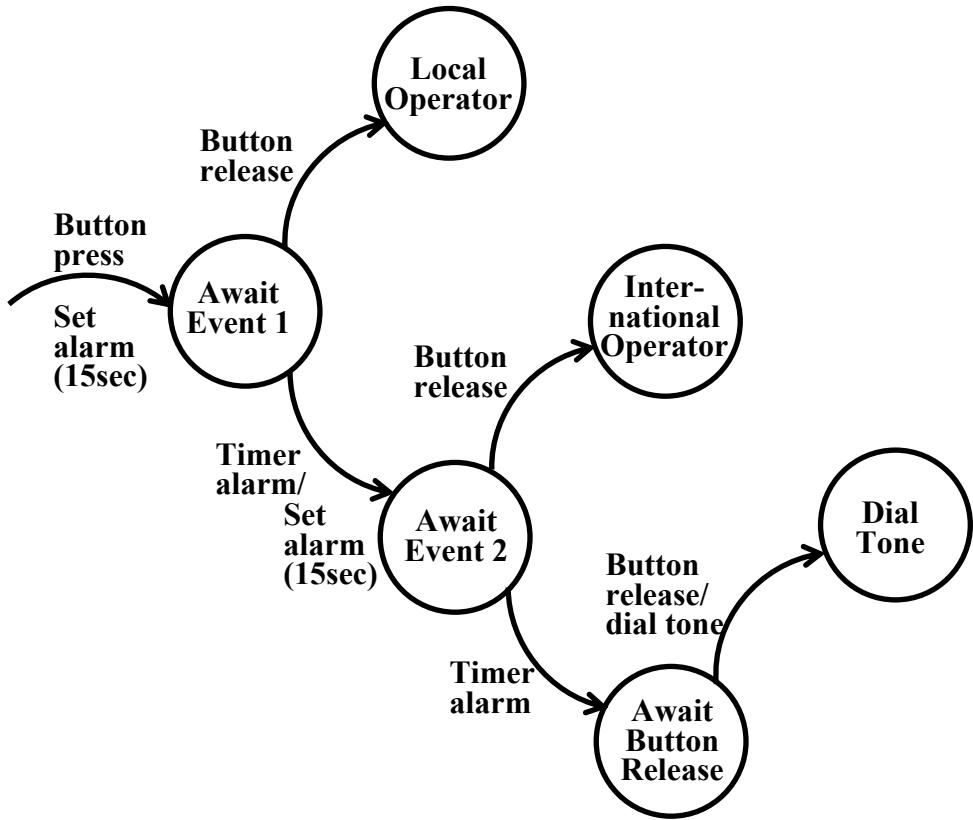
A delay constraint between two events is one where after an event occurs, a minimum time must elapse before the other event can occur. We had considered the following example of delay constraint in Sec. 1.3: *After a digit is dialed, the next digit should be dialed no sooner than 10 milliseconds*. The EFSM model for it is shown in Fig. 35.10. In Fig. 35.10, if the next digit appears before the alarm, then the beeping sound is produced and the system transits to “*Await Caller On-hook*” state.



**Fig. 35.10 Model of an SS Type of Delay Constraint**

### 2.2.6. Durational Constraint

In case of a durational constraint, an event is required to occur for a specific duration. The example of a durational constraint we had considered in Sec. 1.3 is the following: *If you press the button of the handset for less than 15 seconds it connects to the local operator. If you press the button for any duration lasting between 15 to 30 seconds, it connects to the international operator. If you keep the button pressed for more than 30 seconds, then on releasing it would produce the dial tone.*



**Fig. 35.11 A Model of a Durational Constraint**

The EFSM model for this example is shown in Fig. 35.11. Note that we have introduced two intermediate states “*Await Event 1*” and “*Await Event 2*” to model a durational constraint.

### 3. Exercises

1. Mark the following as True or False. Justify your answer.
  - a. A deadline constraint between two stimuli can be considered to be a behavioural constraint on the environment of the system.
2. Identify and represent the timing constraints in the following air-defense system by means of an extended state machine diagram. Classify each constraint into either performance or behavioral constraint.

Every incoming missile must be detected within 0.2 seconds of its entering the radar coverage area. The intercept missile should be engaged within 5 seconds of detection of the target missile. The intercept missile should be fired after 0.1 Seconds of its engagement but no later than 1 second.

3. Represent a wash-machine having the following specification by means of an extended state machine diagram.

The wash-machine waits for the start switch to be pressed. After the user presses the start switch, the machine fills the wash tub with either hot or cold water depending upon the setting of the HotWash switch. The water filling continues until the high level is sensed. The machine starts the agitation motor and continues agitating the wash tub until either the

preset timer expires or the user presses the stop switch. After the agitation stops, the machine waits for the user to press the startDrying switch. After the user presses the startDrying switch, the machine starts the hot air blower and continues blowing hot air into the drying chamber until either the user presses the Stop switch or the preset timer expires.

4. What is the difference between a performance constraint and a behavioral constraint? Give practical examples of each type of constraint.
5. Represent the timing constraints in a collision avoidance task in an air surveillance system as an extended finite state machine (EFSM) diagram. The collision avoidance task consists of the following activities.
  - The first subtask named radar signal processor processes the radar signal on a signal processor to generate the track record in terms of the target's location and velocity within 100 mSec of receipt of the signal.
  - The track record is transmitted to the data processor within 1 mSec after the track record is determined.
  - A subtask on the data processor correlates the received track record with the track records of other targets that come close to detect potential collision that might occur within the next 500 mSec.
  - If a collision is anticipated, then the corrective action is determined within 10 mSec by another subtask running on the data processor.
  - The corrective action is transmitted to the track correction task within 25 mSec.
6. Consider the following (partial) specification of a real-time system:

The velocity of a space-craft must be sampled by a computer on-board the spacecraft at least once every second (the sampling event is denoted by S). After sampling the velocity, the current position is computed (denoted by event C) within 100 msec, parallelly, the expected position of the space-craft is retrieved from the database within 200 msec (denoted by event R). Using these data, the deviation from the normal course of the space-craft must be determined within 100 msec (denoted by event D) and corrective velocity adjustments must be carried out before a new velocity value is sampled in (the velocity adjustment event is denoted by A). Calculated positions must be transmitted to the earth station at least once every minute (position transmission event is denoted by the event T). Identify the different timing constraints in the system. Classify these into either performance or behavioral constraints. Construct an EFSM to model the system.
7. Construct the EFSM model of a telephone system whose (partial) behavior is described below:

After lifting the receiver handset, the dial tone should appear within 20 seconds. If a dial tone can not be given within 20 seconds, then an idle tone is produced. After the dial tone appears, the first digit should be dialled within 10 seconds and the subsequent five digits within 5 seconds of each other. If the dialling of any of the digits is delayed, then an idle tone is produced. The idle tone continues until the receiver handset is replaced.
8. What are the different types of timing constraints that can occur in a system? Give examples of each.