Then the response is calculated using the M-file *response.m* as follows:

```
>>y = response(num,den,t,u); <enter>
```

where *num* and *den* were specified in Example 2.13. The calculated response is plotted in Figure 2.30. Note the ease with which *response.m* calculates the response to a complicated input function. A more general method of evaluating response of even multi-input, multi-output systems based on the *state-space* approach will be given in Chapter 4. MATLAB (CST) functions use the *state-space* approach for calculating step, impulse, and arbitrary input responses; hence, discussion of these functions will be postponed until Chapter 4.

There are three properties which determine whether a control system is good or bad, namely its *performance, stability*, and *robustness*. We briefly discussed the implications of each of these in Chapter 1 using the car-driver example. Now we are well equipped to define each of these three properties precisely. Let us first consider the performance of a control system.

2.7 Performance

Performance is all about *how successfully* a control system meets its desired objectives. Figure 1.3 showed an example of a closed-loop system's performance in terms of the *maximum overshoot* of the actual output, $y(t)$, from the desired *constant* output, y_d. More generally, the desired output may be a specific function of time, $y_d(t)$. In such a case, the difference between the actual output and the desired output, called *error*, $e(t) = y_d(t) - y(t)$, is an important measure of the control system's performance. If the error, $e(t)$, becomes zero very rapidly, the control system is said to perform very well. However, the error of certain control systems may not exactly reach zero for even very large times. For such systems, another performance parameter is considered important, namely the *steady-state error*, e_{ss}, defined as the value of the error, $e(t)$, in the limit $t \to \infty$. The smaller the *magnitude* of the steady-state error, $|e_{ss}|$, the better a control system is said to perform. There are some performance parameters that indicate the *speed* of a control system's response, such as the *rise time*, T_r, defined as the time taken by the output, $y(t)$, to *first reach* within a specified band, $\pm\varepsilon$, of the *steady-state value*, $y(\infty)$, the *peak time*, T_p, defined as the time taken to reach the first peak (or *maximum overshoot*), and the *settling time*, T_s, defined as the time taken until the output, $y(t)$, finally settles (or comes closer) to within a specified band, $\pm\varepsilon$, of its *steady-state value*, $y(\infty)$.

The performance parameters are usually defined for the step response, $s(t)$, of a system, which implies a *constant* value of the desired output, y_d. Figure 2.31 shows a typical step response for a control system, indicating the various performance parameters. Obviously, we assume that the control system reaches a steady-state value in the limit $t \to \infty$. Not all systems have this property, called *asymptotic stability*, which we will discuss in the next section.

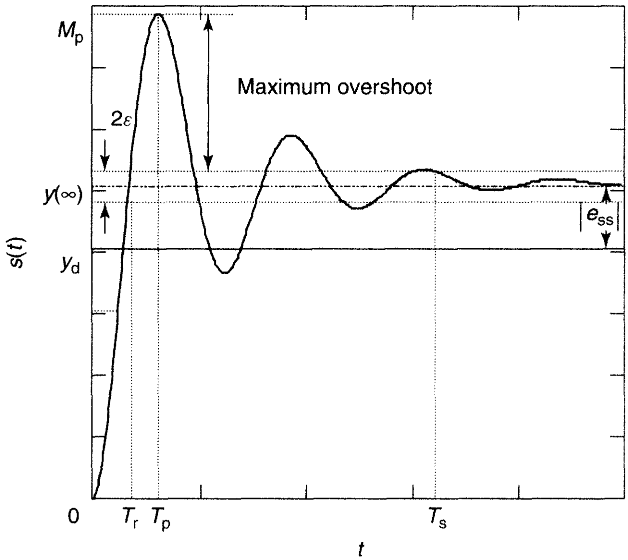

Figure 2.31 Step response, $s(t)$, defining a control system's performance

Note that while the swiftness, or alacrity, with which a system responds to a given input is described by the rise time, T_r, the peak time, T_p, and the settling time, T_s, the departure of the output from its steady-state value is measured by the maximum overshoot, or the *first peak value*, M_p, and the accuracy of the control system in reaching a final desired value, y_d, is indicated by the steady-state error, e_{ss}. For a second order system (such as the one considered in Example 2.12) with damping-ratio, ς, and natural frequency, ω_n, we can find simple expressions for many of the performance parameters *when a unit step input is applied*, given by

$$T_s = 4/(\varsigma\omega_n) \tag{2.125}$$

$$T_p = \pi/[\omega_n(1 - \varsigma^2)^{1/2}] \tag{2.126}$$

and

$$M_p = 1 + \exp\{-\varsigma\pi/(1 - \varsigma^2)^{1/2}\} \tag{2.127}$$

Equations (2.125)–(2.127) can be obtained from the step response, $s(t)$, of a second-order system, using Eq. (2.116). Note that when a successful second order control system reaches its steady-state value *asymptotically* in the limit $t \to \infty$, then for large times, it behaves in a manner quite similar to a first order control system with a pole at $s = -\varsigma\omega_n$, whose output can be expressed as $y(t) = y(\infty)[1 - \exp\{-\varsigma\omega_n t\}]$. Then the settling time can be determined as the time taken when the $y(t)$ settles to within 2 percent of $y(\infty)$, or $0.02 = \exp\{-\varsigma\omega_n T_s\}$, which gives $T_s = -\log(0.02)/(\varsigma\omega_n)$, or $T_s = 4/(\varsigma\omega_n)$, which is the same as Eq. (2.125). In other words, the output, $y(t)$, reaches within 2 percent of the steady-state value, $y(\infty)$ (i.e. $\varepsilon = 0.02$) after four leaps of the *time-constant*, $1/(\varsigma\omega_n)$. Equation (2.127) can be obtained by using the fact that at the maximum overshoot, or the first peak value, M_p, the slope of the step response, $ds(t)/dt$, is zero.

Note that the performance parameters are intimately related to the damping-ratio, ς, and natural frequency, ω_n, of a second order control system. If $\varsigma \geq 1$, a second order system behaves like a *first order* system, with an exponentially decaying step response (see Figure 2.24). Also, you can see from Eqs. (2.125)–(2.127) that the performance parameters determining the swiftness of response (such as the peak time, T_p) and those determining the deviation of the response from the desired steady-state value (such as peak value, M_p) are *contradictory*. In other words, if we try to *increase* the swiftness of the response by suitably adjusting a control system's characteristics (which are given by ς and ω_n for a second order system), we will have to accept *larger* overshoots from the steady-state value, $y(\infty)$, and *vice versa*. How the control system characteristics are modified to achieve a desired set of performance parameters is an essential part of the control system design.

The performance of a control system is determined by the locations of its poles in the Laplace domain. Generally, the poles of a control system may be such that there are a few poles very close to the imaginary axis, and some poles far away from the imaginary axis. As may be clear from examining expressions for step or impulse response, such as Eqs. (2.115) and (2.116), a control system's response is largely dictated by those poles that are the *closest* to the imaginary axis, i.e. the poles that have the *smallest* real part magnitudes. Such poles that dominate the control system's performance are called the *dominant poles*. Many times, it is possible to identify a single pole, or a pair of poles, as the dominant poles. In such cases, a fair idea of the control system's performance can be obtained from the damping and natural frequency of the dominant poles, by using Eqs. (2.125)–(2.127).

The *steady-state error*, e_{ss}, to an *arbitrary input* is an important measure of control system performance. Consider a general single-input, single-output closed-loop system shown in Figure 2.32, where $G(s)$ and $H(s)$ are the transfer-functions of the plant and the *controller* (also called *compensator*), respectively. Such a closed-loop control system is said to have the controller, $H(s)$, in *cascade* (or *series*) with the plant, $G(s)$. (Another closed-loop configuration is also possible in which $H(s)$ is placed in the *feedback path* of (or in *parallel* with) $G(s)$.) The controller applies an input, $U(s)$, to the plant based upon the error, $E(s) = Y_d(s) - Y(s)$. We saw in Chapter 1 an example of how a controller performs the task of controlling a plant in a closed-loop system by ensuring that the plant output, $y(t)$, becomes as close as possible to the desired output, $y_d(t)$, as quickly

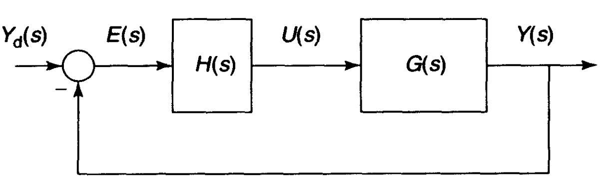

Figure 2.32 A single-input, single-output feedback control system with controller transfer function, $H(s)$, and plant transfer function, $G(s)$

as possible. In any case, a successful control system must bring $y(t)$ very close to $y_d(t)$ when time t becomes very large, i.e. *in the limit that time tends to infinity* ($t \to \infty$). Such a system is called a *tracking system*, because its output, $y(t)$, continuously *tracks* a changing desired output, $y_d(t)$. Examples of tracking systems are a telescope tracking a comet, an antenna tracking a satellite, a missile tracking an aircraft, a rifle-shooter tracking a pigeon, etc. The *error* ($e(t) = y_d(t) - y(t)$) which persists in the limit $t \to \infty$ is called the *steady-state error*, e_{ss}. Obviously, the closed-loop system should first be *able* to reach a *steady-state* (i.e. its response, $y(t)$, must be *finite* and *constant* in the limit $t \to \infty$) before its steady-state error can be defined (it is like saying *you should first be able to stand, before I can measure your height*). An *unstable* system cannot reach a steady-state; therefore, there is no point in talking about steady-state error of unstable systems. We will discuss later what are the precise requirements for stability, but at present let us confine our discussion to *stable closed-loop systems*, which we tentatively define here as those systems in which a *bounded* $y_d(t)$ leads to a *bounded* $y(t)$, for all values of t.

Going back to Figure 2.32, we can see that the Laplace transforms of the output, $y(t)$, desired output, $y_d(t)$, input, $u(t)$, and error, $e(t)$, are given by $Y(s)$, $Y_d(s)$, $U(s)$, and $E(s)$, respectively. Then the steady-state error is expressed as

$$e_{ss} = \lim_{t\to\infty} \mathrm{e}(t) = \lim_{t\to\infty} \mathcal{L}^{-1}(E(s)) \tag{2.128}$$

However, we can avoid evaluating the inverse Laplace transform of $E(s)$ to calculate the steady-state error if we can utilize an important property of the Laplace transform, namely the *final value theorem* given by Eq. (2.70), which yields the following result:

$$e_{ss} = \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) \tag{2.129}$$

Of course, Eq. (2.129) requires that the limit of $e(t)$ when $t \to \infty$ (or of $sE(s)$ when $s \to 0$) must exist. Looking at the block-diagram of Figure 2.32, we can express $E(s)$ as follows:

$$E(s) = Y_d(s) - Y(s) = Y_d(s) - G(s)U(s) = Y_d(s) - G(s)H(s)E(s) \tag{2.130}$$

Thus, we can write

$$E(s) = Y_d(s)/[1 + G(s)H(s)] \tag{2.131}$$

On substituting Eq. (2.131) into Eq. (2.129), we get

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} sY_d(s)/[1 + G(s)H(s)] \tag{2.132}$$

Equation (2.132) implies that the steady-state error, e_{ss}, depends not only upon the two transfer functions, $G(s)$ and $H(s)$, but also on the desired output, $Y_d(s)$.

Example 2.16

Consider the closed-loop system of Figure 2.32 with $G(s) = (2s^2 + 5s + 1)/(s^2 + 2s + 3)$ and $H(s) = K$, where K is a constant. Let us determine the steady-state error of this system if the desired output, $y_d(t)$ is (a) a unit step function, $u_s(t)$, and (b) a unit ramp function, $r(t) = t \cdot u_s(t)$. If $y_d(t) = u_s(t)$ then $Y_d(s) = 1/s$ (see Table 2.1). Hence, the steady-state error is given by Eq. (2.132) as

$$\begin{aligned} e_{ss} &= \lim_{s\to 0} sY_d(s)/[1 + G(s)H(s)] = \lim_{s\to 0} s(1/s)/[1 + KG(s)] \\ &= 1/[1 + K \lim_{s\to 0} G(s)] \end{aligned} \tag{2.133}$$

where $\lim_{s\to 0} G(s)$ is called the *DC gain of* $G(s)$, because it is a property of the system in the limit $s \to 0$, or frequency of oscillation, $\omega \to 0$, in the frequency response $G(i\omega)$ – something like the *direct current* which is the limiting case of *alternating current* in the limit $\omega \to 0$. Here $\lim_{s\to 0} G(s) = 1/3$. Therefore, the steady-state error to unit step function is

$$e_{ss} = 1/(1 + K/3) = 3/(3 + K) \tag{2.134}$$

The CST of MATLAB provides a useful command called *dcgain* for calculating the DC gain of a transfer function, which is used as follows:

```
>>sys= tf(num,den); dcgain(sys) <enter>
```

where *sys* is the name of the system's transfer function calculated using the LTI object *sys*, and *num* and *den* are the numerator and denominator polynomial coefficients (in decreasing powers of s), respectively, of the system's transfer function. This command is quite useful when the transfer function is too complicated to be easily manipulated by hand.

Note that the DC gain of the closed-loop transfer function in Figure 2.32, $G(s)H(s)/[1 + G(s)H(s)]$, is also the steady-state value of the output, $y(t)$, when the desired output is a unit step function. Hence, the steady-state error to a step desired output is nothing but $e_{ss} = 1 -$ DC gain of the *closed-loop* transfer function.

The steady-state error given by Eq. (2.134) can be decreased by making the controller gain, K, large. However, for any finite value of K, we will be left with a non-zero steady-state error. Also, there is a physical limit upto which K (and the resulting input, $U(s)$) can be increased; the larger the value of K, the greater will be the control input, $U(s)$, which increases the *cost* of controlling the system. Hence, this closed-loop system is not very attractive for tracking a desired output which changes by a step.

If $y_d(t) = r(t)$, then noting that $r(t)$ is the time-integral of $u_s(t)$, we can get the Laplace transform, $Y_d(s)$ from the *real integration* property (Eq. (2.68)) as follows:

$$Y_d(s) = \mathcal{L}(r(t)) = \mathcal{L}\left(\int_{-\infty}^{t} u_s(t)\,dt\right) = 1/s^2 \tag{2.135}$$

Hence, the steady-state error is given by

$$e_{ss} = \lim_{s\to 0} sY_d(s)/[1 + G(s)H(s)] = \lim_{s\to 0} s(1/s^2)/[1 + KG(s)]$$
$$= 1/[\lim_{s\to 0} s + sKG(s)] = \infty \tag{2.136}$$

Thus, the steady-state error of the present closed-loop system is *infinite* when the desired output is a ramp function, which is clearly unacceptable. An example of tracking systems whose desired output is a ramp function is an *antenna* which is required to track an object moving at a *constant velocity*. This calls for the antenna to move at a *constant angular velocity*, c. Then the desired output of the antenna is $y_d(t) = c \cdot r(t)$.

Let us see what can be done to reduce the steady-state error of system in Example 2.16 when the desired output is either a unit step or a ramp function.

Example 2.17

In a control system, we can change the controller transfer function, $H(s)$, to meet the desired objectives. This process is called *control system design*. From Example 2.16, it is clear that $H(s) = K$ is a *bad design* for a closed-loop tracking system when the desired output is changing like a step, or like a ramp. If we can make the steady-state error to a ramp function *finite* by somehow changing the system, the steady-state error to a step function will automatically become *zero* (this fact is obvious from Eqs. (2.133) and (2.136)). Let us see what kind of controller transfer function, $H(s)$, will make the steady-state error to a ramp function finite (or possibly zero). For $Y_d(s) = 1/s^2$, the steady-state error is

$$e_{ss} = \lim_{s\to 0} s(1/s^2)/[1 + G(s)H(s)]$$
$$= \lim_{s\to 0}(s^2 + 2s + 3)/s[s^2 + 2s + 3 + (2s^2 + 5s + 1)H(s)]$$
$$= 3/[\lim_{s\to 0} sH(s)] \tag{2.137}$$

If we choose $H(s) = K/s$, then Eq. (2.137) implies that $e_{ss} = 3/K$, which is a finite quantity. If $H(s) = K/s^2$ then $e_{ss} = 0$ from Eq. (2.137). For both the choices of $H(s)$, the steady-state error is zero when $y_d(t) = u_s(t)$. The choice $H(s) = K/s^2$ thus makes the steady-state error zero for both step and ramp functions.

Note that for the closed-loop system of Figure 2.32, the closed-loop transfer function, $Y(s)/Y_d(s)$, can be derived using Eq. (2.131) as follows:

$$Y(s) = G(s)H(s)E(s) = G(s)H(s)Y_d(s)/[1 + G(s)H(s)] \tag{2.138}$$

or

$$Y(s)/Y_d(s) = G(s)H(s)/[1 + G(s)H(s)] \tag{2.139}$$

For single-input, single-output systems such as that shown in Figure 2.32, we can calculate the closed-loop response, $y(t)$, to a specified function, $y_d(t)$, applying the inverse Laplace transform to Eq. (2.139).

Example 2.18

Let us calculate the response of the closed-loop system in Example 2.17 to a unit ramp function, $y_d(t) = r(t)$ and zero initial conditions, when (a) $H(s) = 1/s$, and (b) $H(s) = 1/s^2$. Using Eq. (2.139) for $H(s) = 1/s$ we can write

$$\begin{aligned} Y(s)/Y_d(s) &= [(2s^2+5s+1)/(s^2+2s+3)](1/s)/ \\ &\quad [1+\{(2s^2+5s+1)/(s^2+2s+3)\}(1/s)] \\ &= (2s^2+5s+1)/(s^3+4s^2+8s+1) \end{aligned} \tag{2.140}$$

or

$$\begin{aligned} Y(s) &= (2s^2+5s+1)Y_d(s)/(s^3+4s^2+8s+1) \\ &= (2s^2+5s+1)(1/s^2)/(s^3+4s^2+8s+1) \end{aligned} \tag{2.141}$$

Equation (2.141) can be expressed in a partial fraction expansion as follows:

$$Y(s) = k_1/(s-p_1) + k_2/(s-p_2) + k_3/(s-p_3) + k_4/s + k_5/s^2 \tag{2.142}$$

where p_1, p_2, and p_3 are the poles of the closed-loop transfer function, $(2s^2 + 5s + 1)/(s^3 + 4s^2 + 8s + 1)$, and k_1, k_2, and k_3 are the corresponding residues. k_4 and k_5 are the residues due to the ramp function, $Y_d(s) = 1/s^2$. We know that the poles of the closed-loop transfer function are *distinct* (i.e. not repeated) because we used the Control System Toolbox (CST) command *damp* to get the poles as follows:

```
>>damp([1 4 8 1]) <enter>

Eigenvalue          Damping   Freq. (rad/sec)
-0.1336             1.0000    0.1336
-1.9332+1.9355i     0.7067    2.7356
-1.9332-1.9355i     0.7067    2.7356
```

Note that the closed-loop system has a real pole, -0.1336, and a pair of complex conjugate poles, $-1.9332 + 1.9355i$, and $-1.9332 - 1.9355i$. The residues of the partial fraction expansion (Eq. (2.142)) can be calculated using MATLAB intrinsic command *residue* as follows:

```
>>num=[2 5 1]; den=conv([1 0 0],[1 4 8 1]); <enter>

>>[k,p,c]=residue(num,den) <enter>
```

```
k =
 -0.0265-0.1301i
 -0.0265+0.1301i
 -2.9470
  3.0000
     0

p =
 -1.9332+1.9355i
 -1.9332-1.9355i
 -0.1336
     0
     0

c =
    []
```

The roots of the denominator polynomial of Eq. (2.141) are contained in the vector p, while the vector k contains the corresponding residues of Eq. (2.142). The direct term c is a *null vector*, because the numerator polynomial is of a degree *smaller* than the denominator polynomial in Eq. (2.141). Taking the inverse Laplace transform of Eq. (2.142), we can express $y(t)$ as follows:

$$y(t) = k_1 \exp(p_1 t) + k_2 \exp(p_2 t) + k_3 \exp(p_3 t) + k_4 + k_5 t; \quad (t \geq 0) \quad (2.143)$$

The error $e(t) = y_d(t) - y(t)$, where $y_d(t) = r(t)$, and $y(t)$ is given by Eq. (2.143) is plotted in Figure 2.33 using MATLAB as follows (we could also have

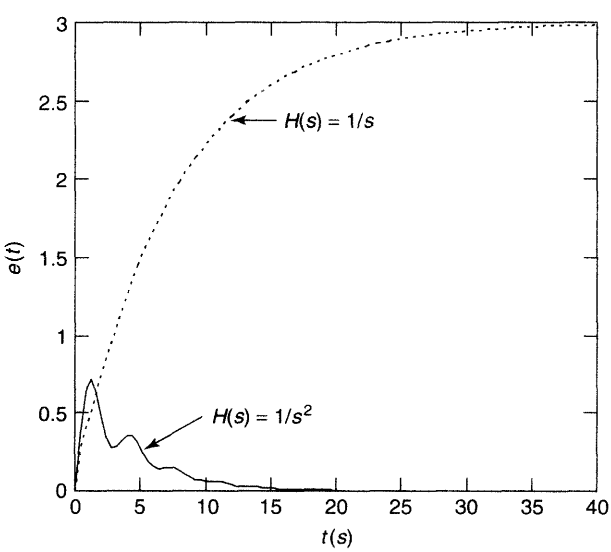

Figure 2.33 Error, $e(t) = y_d(t) - y(t)$, for the closed-loop systems of Example 2.18 when $y_d(t) = r(t)$

obtained $y(t)$ directly by using the M-file *response.m* listed in Table 2.4 by specifying a ramp input):

```
>>t=0:0.4:40; y=k(1)*exp(p(1)*t)+k(2)*exp(p(2)*t)+k(3)*exp(p(3)*t)
  +k(4)+k(5)*t; e=t-y <enter>

>>plot(t,e)
```

Similarly, when $H(s) = 1/s^2$, the closed-loop transfer function is

$$Y(s)/Y_d(s) = (2s^2 + 5s + 1)/(s^4 + 2s^3 + 5s^2 + 5s + 1) \tag{2.144}$$

and the inverse Laplace transform applied to $Y(s)$ with $Y_d(s) = 1/s^2$ yields (you may verify using MATLAB)

$$y(t) = k_1 \exp(p_1 t) + k_2 \exp(p_2 t) + k_3 \exp(p_3 t) + k_4 \exp(p_4 t) + k_5 + k_6 t;\ (t \geq 0) \tag{2.145}$$

where $p_1 = -0.3686 + 1.9158i$, $p_2 = -0.3686 - 1.9158i$, $p_3 = -1.0$, $p_4 = -0.2627$, $k_1 = -0.1352 - 0.1252i$, $k_2 = -0.1352 + 0.1252i$, $k_3 = -0.6667$, $k_4 = 0.9371$, $k_5 = 0$, $k_6 = 0$. The error, $e(t) = y_d(t) - y(t)$, for $H(s) = 1/s^2$ is also plotted in Figure 2.33. Note that $e_{ss} = 3$ for $H(s) = 1/s$, and $e_{ss} = 0$ for $H(s) = 1/s^2$, as expected from Example 2.17.

We have seen in Examples 2.16–2.18 that for a plant transfer function, $G(s)$, of a particular form, the controller transfer function, $H(s)$, must have either one or two poles *at the origin* $(s = 0)$ in order to reduce the closed-loop error due to ramp function. Precisely how many poles $H(s)$ should have to reduce the steady-state error of a closed-loop system to a particular desired output, $y_d(t)$, depends upon the plant transfer function, $G(s)$, and $y_d(t)$. When $y_d(t)$ is a ramp function, Eq. (2.132) implies that

$$\begin{aligned} e_{ss} &= \lim_{s\to 0} s(1/s^2)/[1 + G(s)H(s)] = 1/[\lim_{s\to 0}(s + sG(s)H(s))] \\ &= 1/[\lim_{s\to 0} sG(s)H(s)] \end{aligned} \tag{2.146}$$

Clearly, if we want zero steady-state error when desired output is a ramp function, then Eq. (2.144) requires that $\lim_{s\to 0} sG(s)H(s) = \infty$, which is possible only if the transfer function $G(s)H(s)$ has *two or more* poles at the origin, $s = 0$. Since $G(s)$ in Examples 2.17 had *no poles* at the origin, we had to choose $H(s)$ with *two* poles at the origin (i.e. $H(s) = 1/s^2$) to make $e_{ss} = 0$ when $y_d(t) = r(t)$. Classical control assigns a *type* to a closed-loop system of Figure 2.32 according to *how many* poles the transfer function, $G(s)H(s)$, has at the origin. Thus, a *type 1* system has exactly one pole of $G(s)H(s)$ at origin, a *type 2* system has exactly *two* poles of $G(s)H(s)$ at the origin, and so on. The transfer function, $G(s)H(s)$, is called the *open-loop transfer function* of the system in Figure 2.32, because $Y(s)/Y_d(s) = G(s)H(s)$ if the feedback loop is broken (or *opened*). We know from the real integration property of the Laplace transform (Eq. (2.68)) that a pole at the origin results from a time-integration. Hence, it is said in