

## Lecture # 4

### Block Diagrams

Generally it is more convenient to describe a function by its pictorial representation. For example the space remaining and used in a computer hard-disk can be easily recognized by a pi-chart display rather than numerical data. In order to describe and understand a physical process it is more convenient to provide a pictorial representation of different stages through which input is synthesized. Control systems consist of multiple subsystems, each can be represented by its single individual transfer function in a block. Since the response of a single transfer function can be calculated, the response of multiple subsystems can be calculated on the basis of a single block transfer function as shown in Figure (1) with a terminating (inward) arrow to the block represents an input signal  $R(s)$  and originating (outward) arrow from the block represents the output quantity  $C(s)$ .

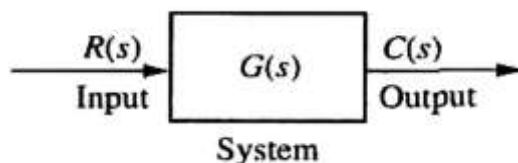


Figure 1: Block Diagram Representation

Mathematically the output  $C(s)$  is  $R(s)$  multiplied by the transfer function  $G(s)$  that is:  $C(s) = R(s)G(s)$ . A feed-forward or a forward block  $G(s)$  transmit signal from input to output direction whereas; a feedback block  $H(s)$  transmit signal from output to input direction. Multiple subsystems are represented in two ways: as block diagrams and as signal-flow graphs. Although neither representation is limited to a particular analysis and design technique, block diagrams are usually used for  $s$ -domain analysis and design, and signal-flow graphs for state-space analysis and design.

### Summing Junction

When multiple subsystems are interconnected, a few more schematic elements must be added to the block diagram. These additional elements are summing junctions and takeoff points as shown in Fig (7.2).

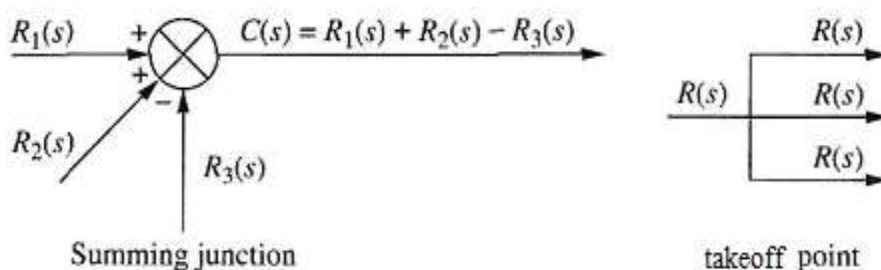


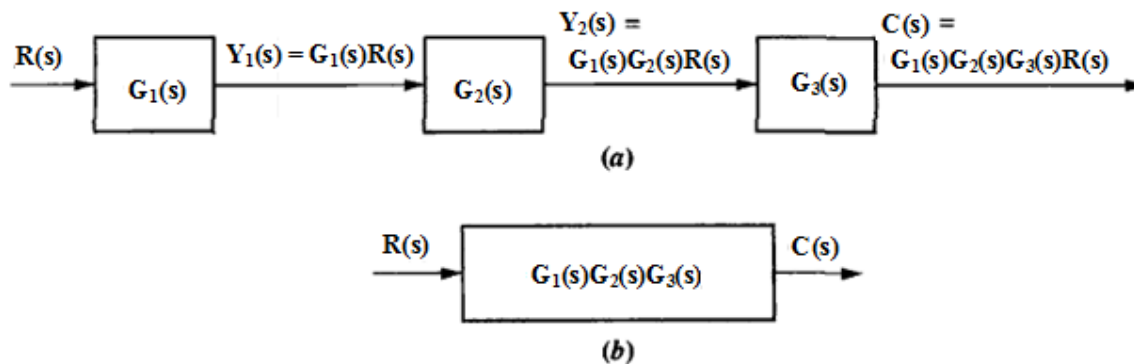
Figure 2: Summing Junction and Takeoff Point

The characteristic of the summing junction is that the output signal,  $C(s)$ , is the algebraic sum of the input signals. A takeoff point, distributes the input signal,  $R(s)$  to numerous points.

### Cascade Form

Blocks are said to be connected in cascade if the output of the preceding block appears as an input to the following block. Thus a cascade arrangement of blocks is viewed as blocks in series (one after the other). Figure (3a) shows an example of cascaded subsystems in which three subsystems with their transfer functions  $G_1(s)$ ,  $G_2(s)$  and  $G_3(s)$  are connected in cascade so that the equivalent representation is a single block as shown in Figure (3b) with an overall transfer function,  $G(s)$  as:

$$G(s) = \frac{C(s)}{R(s)} = G_1(s)G_2(s)G_3(s) \quad 1$$



**Figure 3: Blocks in Cascade**

Thus the overall transfer function of blocks connected in cascade is the product of the transfer function of the individual subsystem block.

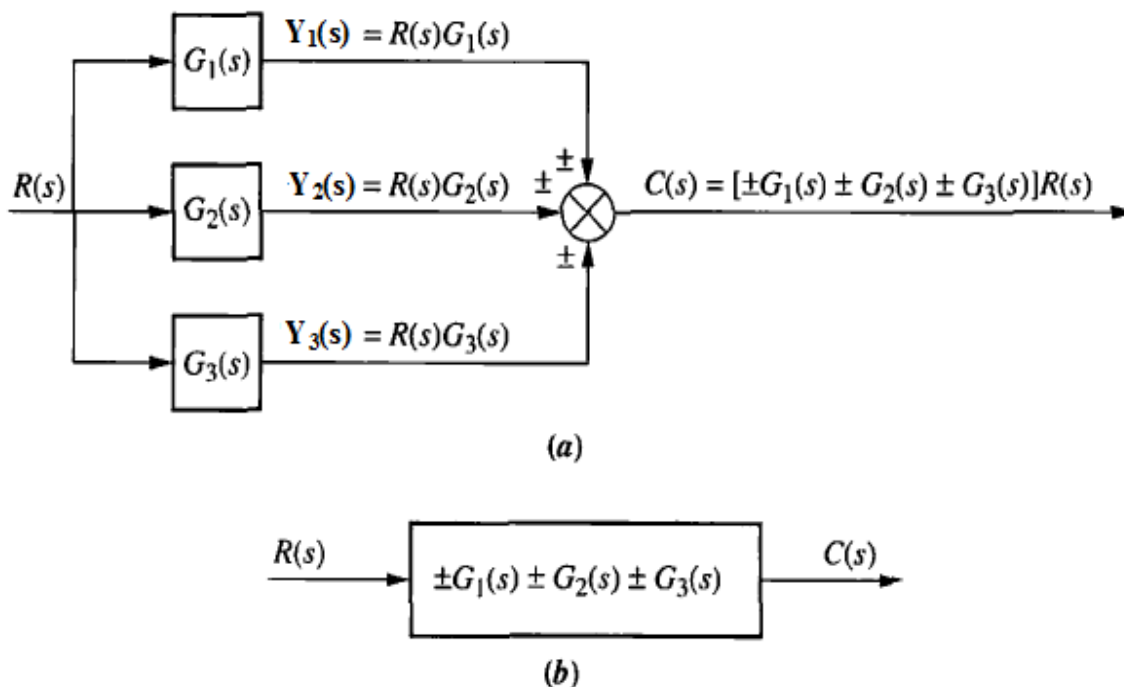
### Parallel Form

Blocks are said to be connected in parallel if they derive their inputs from a common source, whereas the output of each block is different depending on their transfer function. Thus a summing junction at the output is necessary. Figure (4a) shows an example of arrangement of three subsystems with their transfer functions  $G_1(s)$ ,  $G_2(s)$  and  $G_3(s)$  connected in parallel. Knowing the output of each subsystem, we can find the equivalent transfer function. From Figure (4b), at the summing junction the output of each block  $Y_1(s)$ ,  $Y_2(s)$  and  $Y_3(s)$  are added so that:

$$C(s) = \pm Y_1(s) \pm Y_2(s) \pm Y_3(s)$$

$$C(s) = \pm R(s)G_1(s) \pm R(s)G_2(s) \pm R(s)G_3(s)$$

Or 
$$G(s) = \frac{C(s)}{R(s)} = \pm G_1(s) \pm G_2(s) \pm G_3(s) \quad 2$$

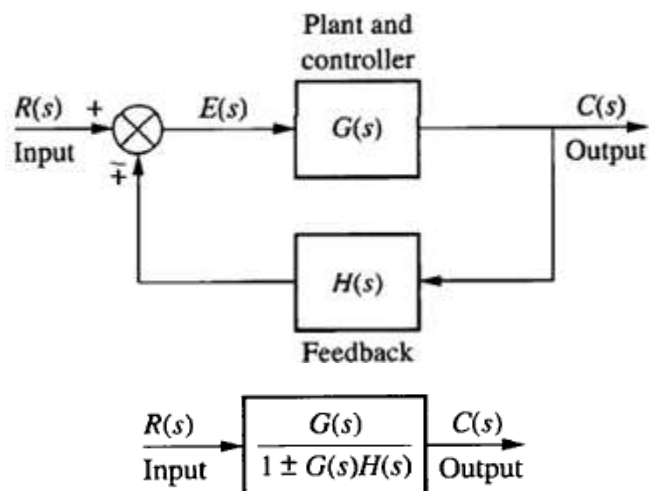


**Figure 4: Subsystem Blocks in Parallel**

Thus when subsystems are in parallel, the combined transfer function placed in a single block is the sum of transfer functions of individual subsystems.

### Feedback Form

All automatic control systems incorporate feedback to modify the input to suit the requirements at the output. Automatic control systems are therefore referred to as feedback control systems. A typical feedback system is shown in Figure (5) in which the forward transfer function  $G(s)$  is obtained from the cascade combination of the controller and the plant, whereas the transfer function  $H(s)$  represent the transfer function of cascaded feedback element.



**Figure 5: Simplification of a Feedback Control System**

At the summing junction:  $E(s) = R(s) \pm C(s)H(s)$

$$\text{Or } C(s) \frac{1}{G(s)} = R(s) \pm C(s)H(s)$$

Simplifying and rearranging yields the overall closed-loop transfer function as:

$$T(s) = \frac{C(s)}{R(s)} = \frac{G(s)}{1 \pm G(s)H(s)} \quad 3$$

**Important Point:** Thus to obtain a single block representation and overall transfer function, the first step is to simplify those blocks which are directly connected in either cascade or in parallel. The next step is to consider the loop. In a “loop within a loop” situation, the inner loop simplified first followed by the outer loop. The arrows direction must be considered in order to identify feed-forward and feedback blocks.

**Example 1:** Simplify the given block diagram as shown in Figure (6) to a single block and obtain its transfer function.

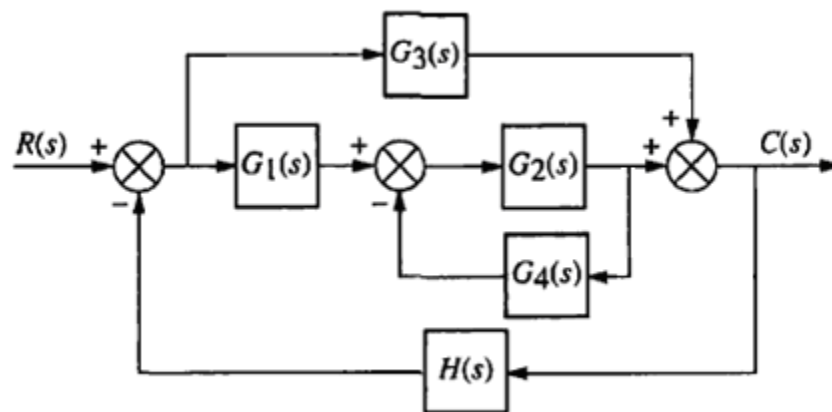


Figure 6

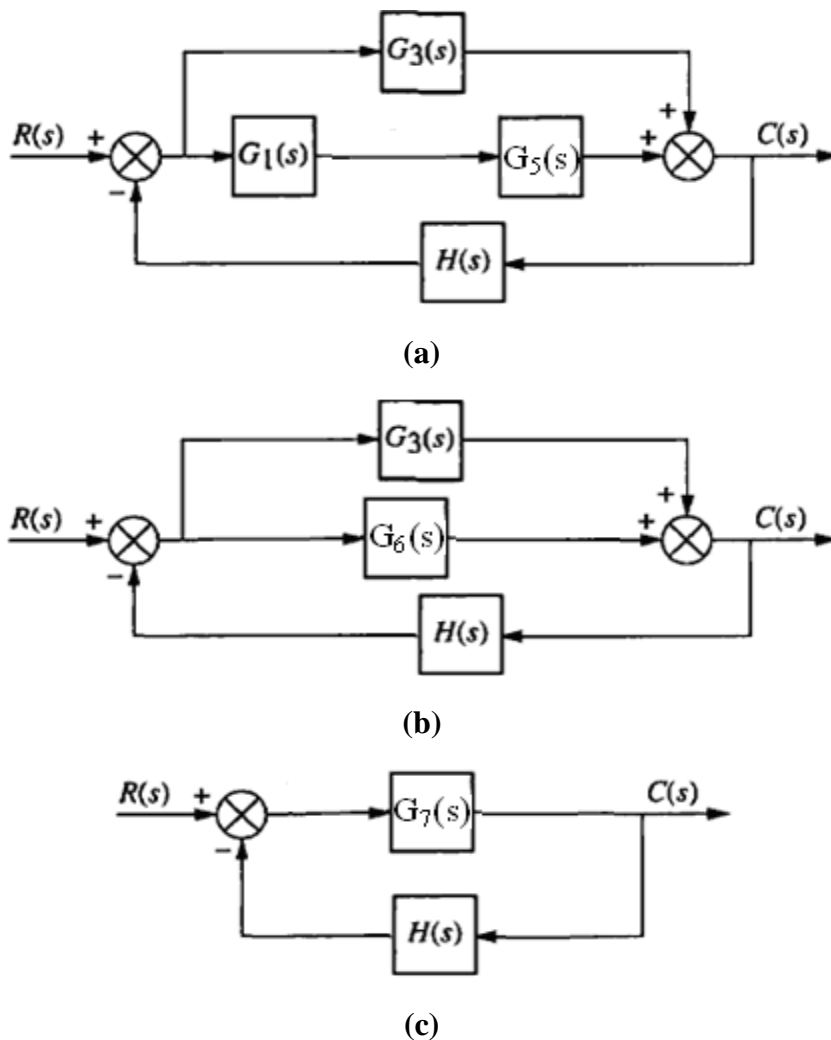
**Solution:** The first step in the simplification is that all those single blocks, which are connected directly in series (cascade) or in parallel are simplified. In Figure (6), however, there is apparently no such situation of blocks directly in parallel or cascade. The next step is to consider the feedback loops. We have a “loop within a loop” situation; the innermost loop is simplified first followed by the outer loops in sequence. The arrows direction must be considered in order to identify feed-forward and feedback blocks. In Figure (6), the loop containing  $G_2(s)$  and  $G_4(s)$  are simplified to a single block  $G_5(s)$  as:

$$G_5(s) = \frac{G_2(s)}{1 + G_2(s)G_4(s)}$$

The resultant system is shown in Figure (7a). It can be seen that the block  $G_1(s)$  is now in cascade (series) with block  $G_5(s)$ , whose transfer functions can be multiplied to obtain the overall transfer function  $G_6(s)$  as:

$$G_6(s) = G_1(s)G_5(s)$$

The resultant block diagram with block  $G_6(s)$  is shown in Figure (7b).



**Figure 7: Step-by-step Block Diagram Reduction**

In Figure (7b) the blocks  $G_3(s)$  and  $G_6(s)$  form a parallel combination, so that their transfer functions are added to result in  $G_7(s)$  that is:

$$G_7(s) = G_3(s) + G_6(s) = G_3(s) + G_1(s)G_5(s)$$

Or

$$G_7(s) = G_3(s) + G_1(s) \frac{G_2(s)}{1 + G_2(s)G_4(s)}$$

The resultant system is shown in Fig (7c), which a simplified canonical form.

$$T(s) = \frac{G_7(s)}{1 + H(s)G_7(s)}$$

Substituting the values of the  $G_7(s)$ :

$$T(s) = \frac{G_3(s) + G_1(s) \left[ \frac{G_2(s)}{1 + G_2(s)G_4(s)} \right]}{1 + H(s) \left[ G_3(s) + G_1(s) \frac{G_2(s)}{1 + G_2(s)G_4(s)} \right]}$$

$$T(s) = \frac{G_3(s)[1 + G_2(s)G_4(s)] + G_1(s)G_2(s)}{1 + G_2(s)G_4(s) + H(s)[G_3(s)(1 + G_2(s)G_4(s)) + G_1(s)G_2(s)]}$$

$$T(s) = \frac{G_3(s) + G_2(s)G_3(s)G_4(s) + G_1(s)G_2(s)}{1 + G_2(s)G_4(s) + G_1(s)G_2(s)H(s) + G_3(s)H(s) + G_2(s)G_3(s)G_4(s)H(s)}$$

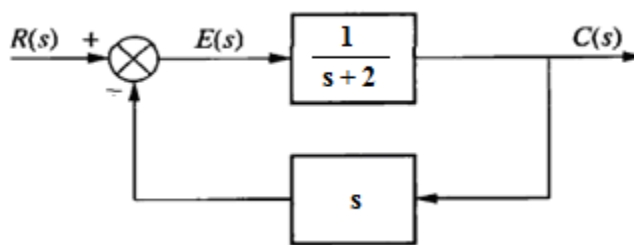
### Converting to Unity Feedback System

In numerous control system studies, it is usually convenient to represent a canonical form as an equivalent unity feedback system. This can greatly simplify control system problems. In unity feedback systems, the forward block has a transfer function  $G(s)$ , whereas the feedback path has a transfer function of unity [ $H(s) = 1$ ], as:

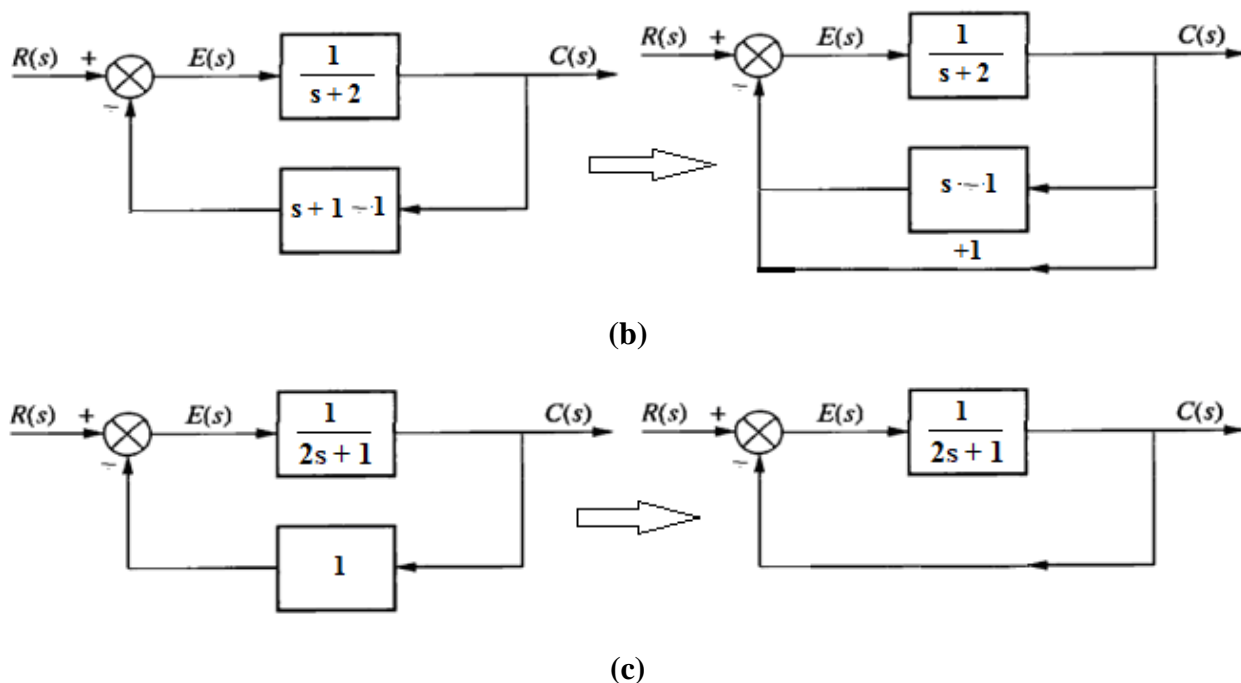
$$T(s) = \frac{C(s)}{R(s)} = \frac{G(s)}{1 \pm G(s)} \quad 4$$

The first step is to add and subtract 1 from the feedback element, leaving +1 as being independent block while combining  $-1$  with the feedback block transfer function. The second step is to decompose the feedback block in two parallel parts whose input  $C(s)$  and terminates at the same summing point in the same form (positive or negative) as that of the given system. One portion is a block whose function is +1 and the other block is a modified function in which  $-1$  has been combined with feedback transfer function in the first step. The separation thus forms two loops, an inner and an outer loop. The third step is to consider the inner loop for obtaining the closed-loop transfer function, leaving the +1 feedback block as it is which forms a unity feedback system.

**Example 2:** Consider a canonical form of negative feedback control system with the feedback element  $s$  as shown in Figure (8a).



(a)



**Figure 8: Converting to Unity Feedback System**

The first step and second step is illustrated in Figure (8b). The third step is to simplify the inner loop leaving the outer loop with 1 as shown in Figure (8c). Thus the transfer function of the system shown in Figure (8c) is:

$$T(s) = \frac{1/(2s+1)}{1+1/(2s+1)} = \frac{1}{2s+2} = \frac{1/2}{s+1}$$

The transfer function of Figure (8a) is:

$$T(s) = \frac{1/(s+2)}{1+(s)/(s+2)} = \frac{1}{2s+2} = \frac{1/2}{s+1}$$

The two are the same.

### Signal-Flow Graphs (SGF)

First developed by S. J. Mason, the SFG shows the flow of information between different subsystems from the input to the output. SFG are an alternative to block diagrams and are much simpler to understand. SFG consists only of branches (transmittance or path gain) and nodes (junctions). A branch links two nodes and may be forward or feedback. A forward branch is recognized by arrow from input side to output, whereas the feedback branch is recognized by arrow from output side to input. A node performs the function of addition of the signals on all incoming branches and the transmission of the total node signal (the sum of all incoming signals) to all outgoing branches. The following three types of nodes are of particular interest in SFG.

**Source nodes (independent nodes):** These represent independent variables and have only outgoing branches; e.g. input nodes.

**Sink nodes (dependent nodes):** These represent dependent variables and have only incoming branches; e.g. output node.

**Mixed nodes (general nodes):** These have both incoming and outgoing branches.

**Path:** A path is defined as connected sequence of branches with their arrows in the same direction. Paths in SFG are recognized as forward and feedback. A forward path between two nodes is one that follows the arrows of successive branches from input to output and in which a node is traversed only once. On the contrary, a feedback path between two nodes is one that follows the arrows of successive branches from output to input and in which a node is traversed only once.

### Mason's Rule

The relationship between an input variable and output variable of a SFG is the net gain or transmittance between the input and output nodes and represents the overall gain or transfer function of the system. Mason's rule allows us to reduce SFG of a block diagram to single closed-loop transfer function  $T(s)$  that requires the application of the Mason's gain formula, which is:

$$T(s) = \frac{\sum P_i \Delta_i}{\Delta} = \frac{P_1 \Delta_1 + P_2 \Delta_2 + \dots}{\Delta} \quad 5$$

Where: 
$$\Delta = 1 - \sum L_a + \sum L_b L_c - \sum L_d L_e L_f + \dots \quad 6$$

And  $\sum L_a$  = algebraic sum of all the possible loop gain

$\sum L_b L_c$  = algebraic sum of product of set of two non-touching loop gain

$\sum L_d L_e L_f$  = algebraic sum of product of set of three non-touching loop gain

Systems can have numerous loops, so that there is a possibility of sets of two, three or more non-touching loops, whose product is then included in Eq (6).  $\Delta_1, \Delta_2, \dots$  are the cofactors of their forward paths  $P_1, P_2, \dots$  respectively. For example,  $\Delta_1$  is the determinant of the remaining sub-graph when the forward path which produces  $P_1$  is removed. In other words  $\Delta_1$  includes the number of leftover loops when a path  $P_1$  is eliminated. The leftover loops are apparent from the resultant sub-graph obtained by removing a forward path. Thus  $\Delta_i$  is equal to unity when the forward path touches all the loops in the SFG (that is the forward path and loops have common nodes) or when the sub-graph contains no loop.

Application of Mason's rule to obtain the overall transfer function requires two steps. In the first step, the given block diagram is converted to SFG, which if possible can be simplified by elimination or absorption of node(s). The second step is the application of Mason's gain

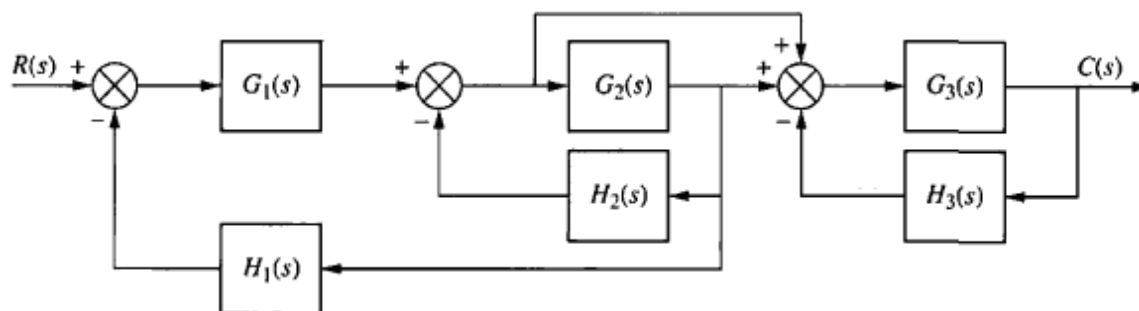


formula. Before applying Mason's gain formula, the components of Mason's formula must be evaluated.

**Points to remember:**

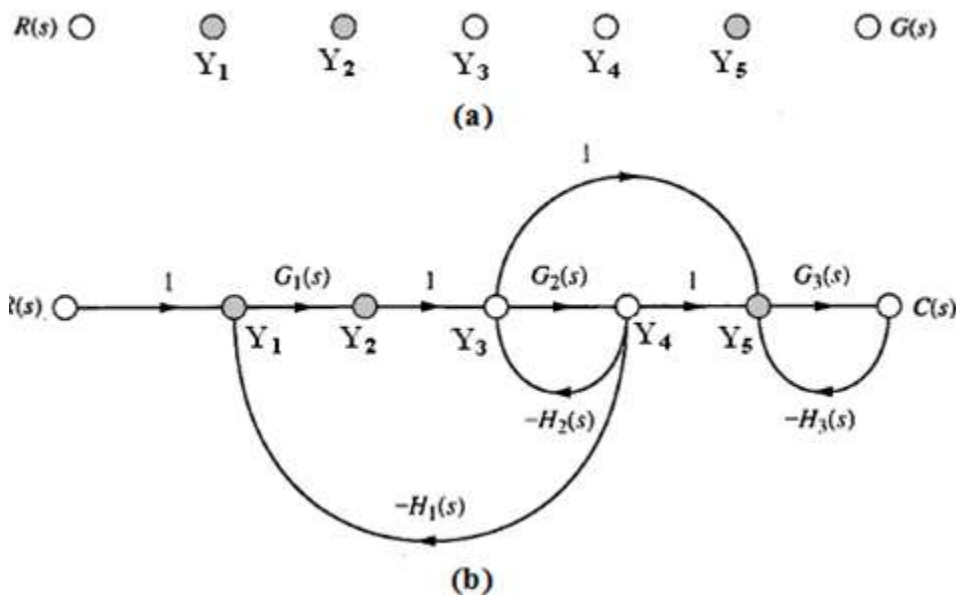
1. The transmittance of each forward path is obtained by multiplying the transmittance along that path with each node traversed once.
2. Loop is identified by a contour following the direction of arrows so as to return at the same node from where we started with nodes in the loop-path traversed once.
3. Loops are touching if they have a node in common, whereas; loops are non-touching if they do not have any node in common.
4. Sometimes in SFG, we can come across self-loop. Self-loop is independent loop with having a single node from which the path originate and terminate.

**Example 3:** Represent the block diagram of Figure (9) by its SFG.



**Figure 9: Block Diagram for Example (7.4)**

**Solution:** The step-by-step approach to the formation of SFG is illustrated in Figure (10).



**Figure 10: Formation of SFG**

As a first step, all possible nodes are laid down representing each summing junction and takeoff points as shown in Figure (10a). The shaded nodes represent summing junctions.

In the second step, the nodes are joined by connecting lines with arrows to indicate the direction of flow of signal, and the transfer function of the respective block representing the transmittance is also indicated. In case there is no transmittance, it is indicated by 1 as in Figure (10b). In the third step the SFG is simplified by absorbing all possible single input and single output nodes. The final SFG of the given block diagram of Figure (9) is shown in Figure (11).

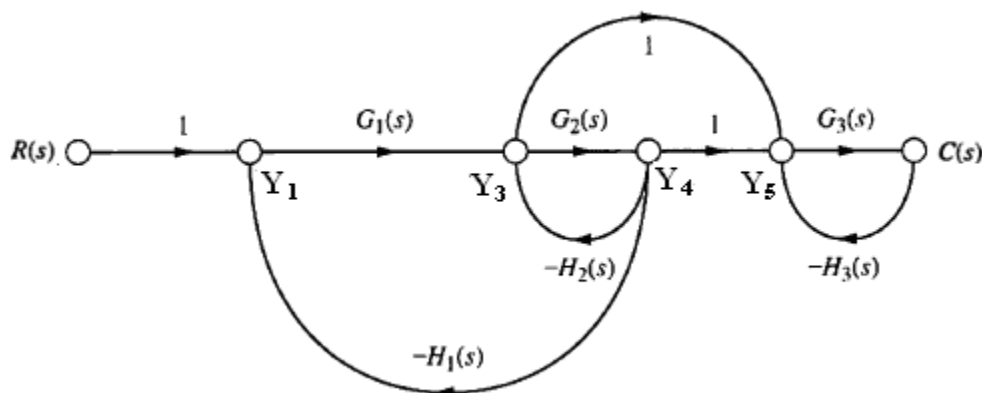


Figure 11: SFG of Block Diagram in Example (3)

**Example 4:** Given the SGF in Figure (12), identify the forward path(s) and loops as components of Mason's formula.

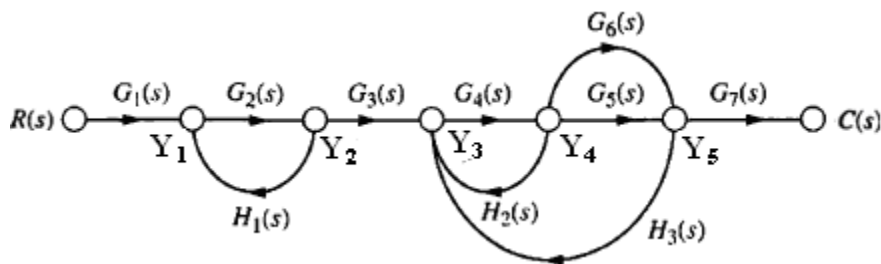


Figure 12

Solution: The main components of the Mason's gain formula are the transmittance of all the forward paths and transmittance of all the individual loops. The components for Mason's formula with reference to Figure (12) are as follows:

1. Independent forward paths:

$$P_1 = G_1 G_2 G_3 G_4 G_5 G_7$$

$$P_2 = G_1 G_2 G_3 G_4 G_6 G_7$$

2. Individual loops:

$$L_1 = G_2H_1$$

$$L_2 = G_4H_2$$

$$L_3 = G_4G_5H_3$$

$$L_4 = G_4G_6H_3$$

3. Product sets of two non-touching loops:

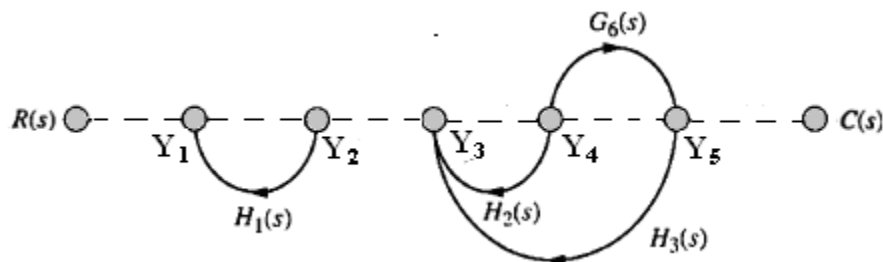
$$L_1L_2: G_2H_1 \text{ and } G_4H_2 \quad [G_2H_1 G_4H_2]$$

$$L_1L_3: G_2H_1 \text{ and } G_4G_5H_3 \quad [G_2H_1 G_4G_5H_3]$$

$$L_1L_4: G_2H_1 \text{ and } G_4G_6H_3 \quad [G_2H_1 G_4G_6H_3]$$

4. There are no three non-touching loops.

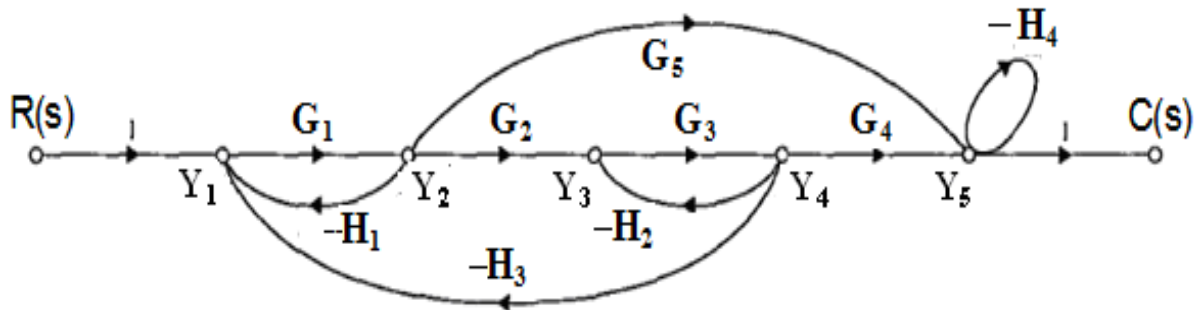
There are two forward paths; there exist  $\Delta_1$  and  $\Delta_2$ , which must also be determined for the use in Mason's formula. For example,  $\Delta_1$  is obtained by referring to the SFG of Figure (12) in which the removed forward path  $P_1$  is indicated by dotted lines and the nodes that are traversed by this forward path are shown shaded. It is thus illustrated in Figure (13) that no loops with solid lines remain and therefore loses the definition of loop. In the expression for  $\Delta$  all the loops containing  $G$ s and  $H$ s are put to zero, resulting in  $\Delta_1 = 1$ .



**Figure 13: SFG when the Forward Path  $P_1$  is removed**

The same procedure can be applied for finding  $\Delta_2$ . The application of Mason's rule is illustrated in the following examples.

**Example 5:** Using Mason's gain formula, obtain the transfer function of a system whose signal flow graph is given in Figure (14).



**Figure 14: SFG of the System with Self-Loop**

**Solution:** The second step is applicable that is components required for the use of Mason's gain formula, which are as follows:

Forward paths:  $P_1 = G_1G_2G_3G_4$

$$P_2 = G_1G_5$$

Individual loops:  $L_1 = -G_1H_1$

$$L_2 = -G_3H_2$$

$$L_3 = -G_1G_2G_3H_3$$

$$L_4 = -H_4 \text{ (Self-loop)}$$

Product sets of two non-touching loops:

$$L_1L_2 = G_1G_3H_1H_2$$

$$L_1L_4 = G_1H_1H_4$$

$$L_2L_4 = G_3H_2H_4$$

$$L_3L_4 = G_1G_2G_3H_3H_4$$

There is one combination of three non-touching loops, whose product set is:

$$L_1L_2L_4 = -G_1G_3H_1H_2H_4$$

Therefore: 
$$\Delta = 1 + G_1H_1 + G_3H_2 + G_1G_2G_3H_3 + H_4 + G_1G_3H_1H_2 + G_1H_1H_4 + G_3H_2H_4 + G_1G_2G_3H_3H_4 + G_1G_3H_1H_2H_4$$

Since all the loops touches the forward path  $P_1$ , therefore:  $\Delta_1 = 1$ . On the other hand the loop formed by the combination of  $G_3$  and  $H_2$  does not touch the forward path  $P_2$ , so that:  $\Delta_2 = 1 + G_3H_2$ . Using:

$$T(s) = \frac{\sum P_i \Delta_i}{\Delta} = \frac{P_1 \Delta_1 + P_2 \Delta_2}{\Delta}$$

$$T(s) = \frac{G_1 G_2 G_3 G_4 + G_1 G_5 (1 + G_3 H_2)}{\Delta}$$

It must be noted that the product set of two and three non-touching loops formed in this example does not contain  $G$ s and  $H$ s that are common in the expressions of the individual loop gains from which the product set is formed. For example, since  $L_1$  and  $L_2$  contain entirely different  $H$ s and  $G$ s, so they are non-touching. Similarly in the product set formed by  $L_1$ ,  $L_2$  and  $L_4$ , of the three non-touching loops there are no  $G$ s and  $H$ s that are common in their expressions, but this cannot be regarded a general case.