Example. Calculate s when t = 5 hours and 30 minutes and d = 500.

Press	See Displayed	
A	0.0000	Runs initialization routine.
5.30 D	5.5000	Time is converted to decimal hours.
500 B	500.0000	Key in the distance.
E C	90.9091	Units per hour.

Example. Calculate t when s = 700 and d = 5000.

Press	See Displayed	
A	0.0000	Runs initialization routine.
700 C	700.0000	Key in the speed in units per hour.
5000 B	5000.0000	Key in the distance
E D	7.0834	7 hours, 8 minutes, and 34 seconds.

Example. Calculate d when s = 60 and t = 74 hours, 42 minutes, and 50 seconds.

Press	See Displayed	
A	0.0000	Runs initialization routine.
60 C	70.0000	Key in the speed in units per hour.
74.4250 D	74.7139	Time is converted to decimal hours.
E B	5229.9722	The answer.

Although flags require valuable memory for setting and unsetting them, they are still handy for program decision making that isn't the result of a direct comparison of the X- and Y-registers.

DSZ as a Flag

By setting the contents of R_s equal to 1, you create your own self-clearing flag using DSZ. When the program executes DSZ, it decrements the contents of R_s, which sets it to zero. Then it tests R_s and, because it is zero, skips two steps before continuing execution (*just as when testing a flag that is set*). The second time the program executes DSZ, the program pointer continues sequentially (*just as when testing a flag that is clear*) because the number in R_s is no longer zero.

Interrupting Your Program

R/S (run /stop) is a special program control key that operates differently from the keyboard than as a program step. As a program step R/S interrupts program execution at an intermediate point, allowing you to key in data, make additional calculations, etc. From the keyboard, R/S will start a program at the position of the active pointer or halt a running program. R/S, however, is also used to control programs differently from what you have learned thus far, so the following information should be studied with care.

To Enter Data

The primary use for R/S in a program (*or subroutine*) is to stop the program in order to allow you to key in data. When a R/S program step is encountered in a running program, the program halts, leaving the pointer at the R/S. By pressing R/S from the keyboard, program execution will continue.

Writing a Program Using R/S to Enter Data. To show you how this works, let's write a program to calculate the cumulative cost of various quantities of differently priced items at a 15% discount.