

## Making a timer without using the timer block

When using MotionWorks+, the "timer" block has both a true and a false port. This may lead one to believe that you can perform one operation *before* time out occurs and a different operation *after* time out. The block isn't really designed to be used that way, however, and problems may occur.

**The situation:** The user wants to jog slowly when a button is pressed, and switch to a higher speed if the button has been held down for a certain length of time (in these examples I used the variable TimeOut).

Figure 1 shows the logical, but incorrect, way.

Jumping out of the false port causes the MP940 to leave the "timer" state and enter another state.

Before the timer can count down, it must re-enter the "timer" state. This means that the actual time for the timer to expire will be more than the TimeOut. There is an additional problem

with this method: if you jump out of the timer when it has only partially counted down, it will not start from the beginning next time. It will have a "remainder" from the previous attempt that it will use instead.

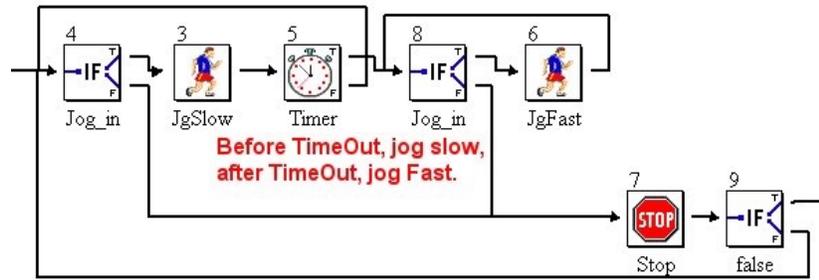


Figure 1 - The Wrong Way

**The fix:** There is a system variable called "mTime". This is a counter (in milliseconds) of the amount of time since the MP940 has been powered up. It increments every scan. While "m" system variables are designed to be used for monitoring, you can set the value of this one directly. By initializing this value to zero, then testing it against the TimeOut variable, a timer can be created.

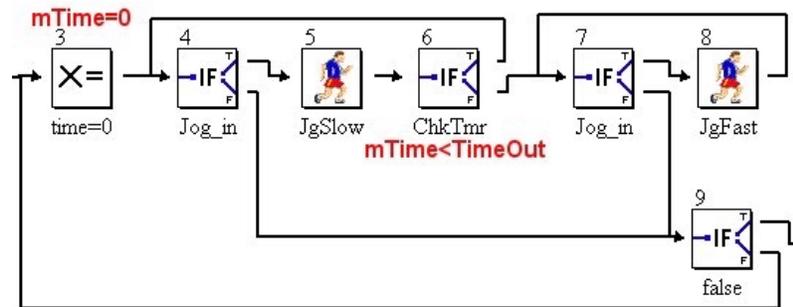


Figure 2 - The Correct Way

Note: When using the "Set Variable" (X=) block, the "variable" dialog does not allow you to select mTime using the dialog box. However, you can type it in directly (make sure you get the capital letters in the right place).

**Multiple Timers:** Obviously, you can only have one timer executed this way at any given time. You could do multiple timers by capturing the mTime value in a variable ([StartTime=mTime](#)) and modifying the check ([mTime<\(TimeOut+StartTime\)](#)).

Changing the mTime system variable will have no effect on "standard" timer blocks.