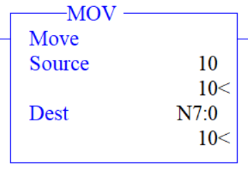
Title: **Move Function** Handout: 3

Course: Introduction to Automation Unit: Introduction of PLC CLO: 4

**Objectives**

1. Student shall develop a knowledge of the move (MOV) function.
2. Student shall strengthen their knowledge of the binary number system.

**Theory**

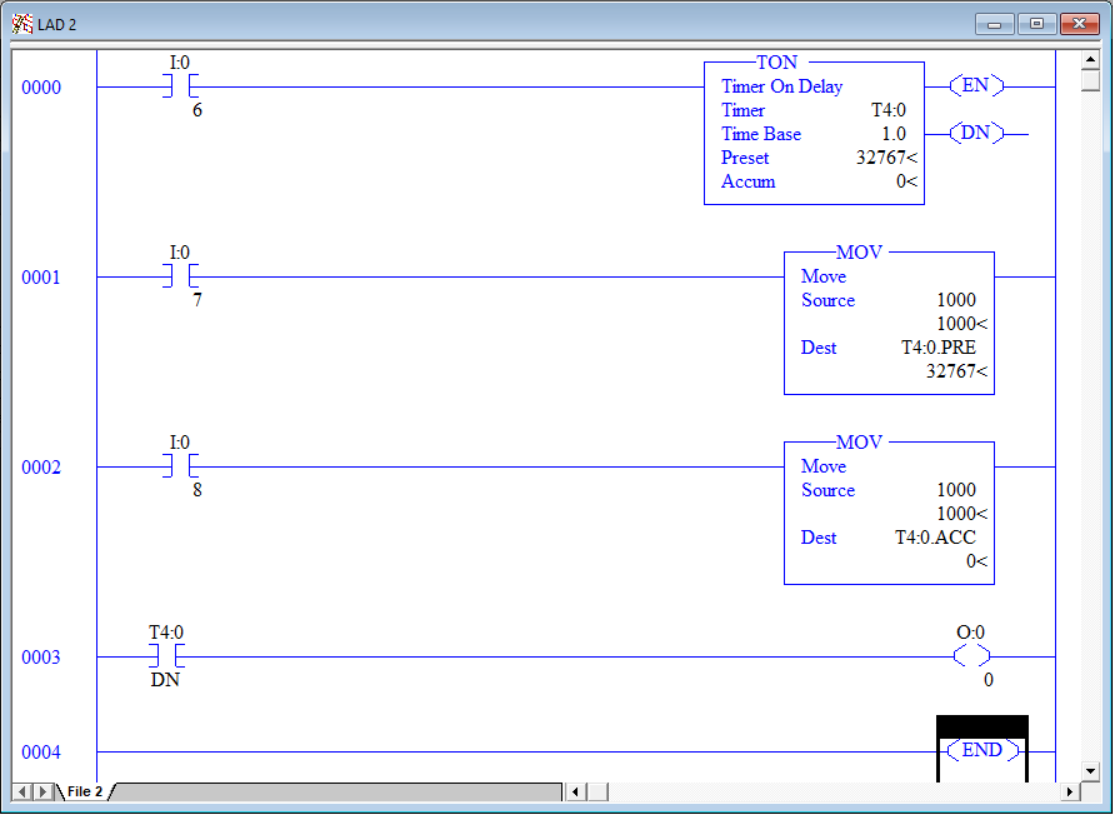
This output instruction transfers the value from one designated location to another. When rung continuity is achieved, the Move function copies a value from the source to the destination each scan. The Source value remains intact and unchanged in its source location. The Source can be an address or a constant of data. The destination is the address that signifies the location to which the data shall be stored. The destination must be an address that can receive the source value. In this example, the constant 10 will be moved in the integer register N7:0 when rung continuity is achieved. If rung continuity is lost, not action is taken.

**Devices**

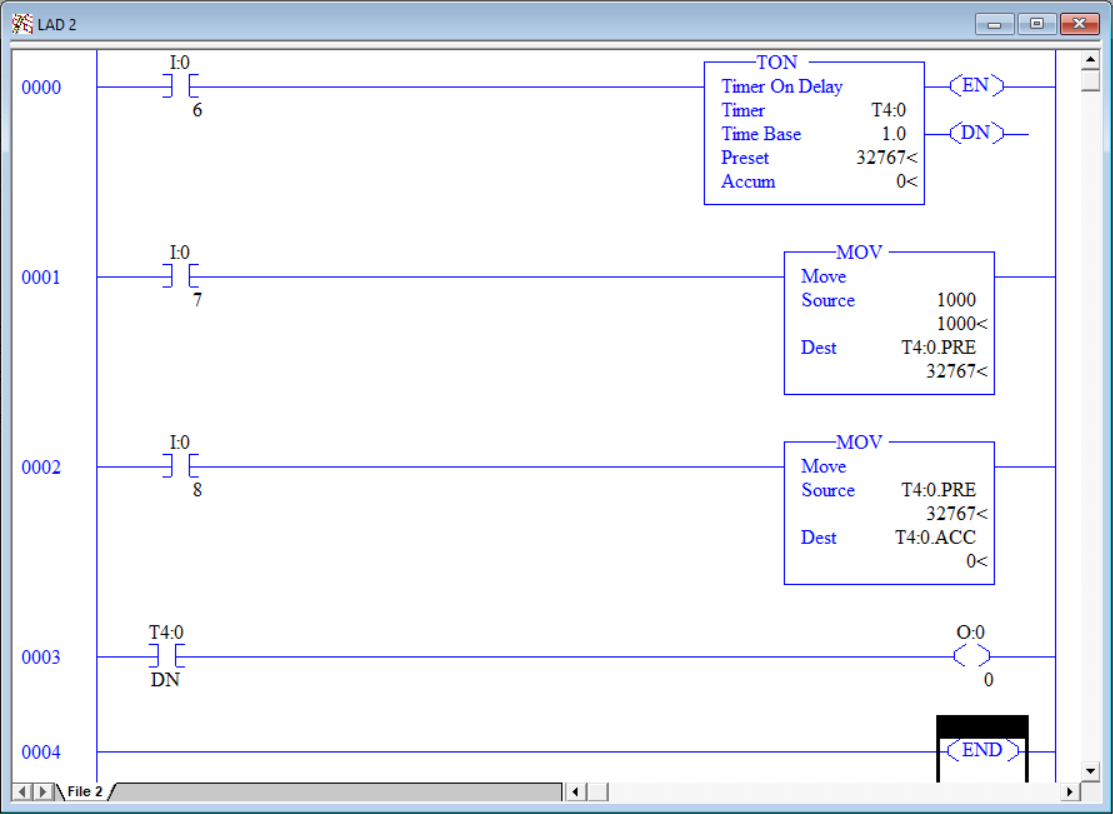
|  |  |  |
| --- | --- | --- |
| Inputs | | |
| *Device* | *Description* | *Symbol* |
| Two-Position Selector Switch | Move Selector | MOV\_SEL |
| NC Pushbutton (PB1) | Reset Timer | RESET |
| NO Pushbutton (PB2) | Move Value to Preset | MOV\_P |
| NO Pushbutton (PB3) | Move Value to Accum | MOV\_A |
| Outputs | | |
| *Device* | *Description* | *Symbol* |
| Green Pilot Light | Timer Done | DONE |

**Instructions**

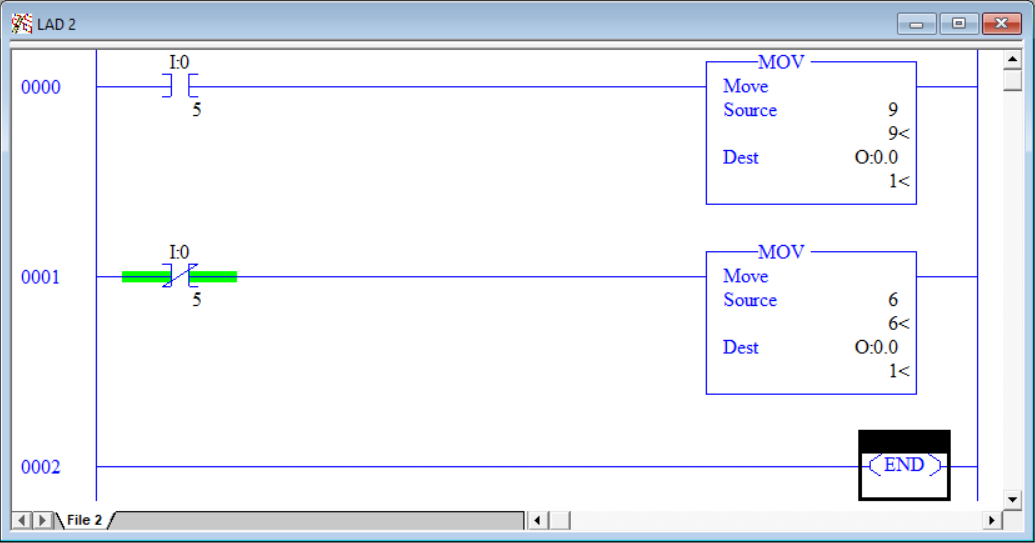
Program the logic should below.



1. Download the program and observe its status online.
2. At its current state, it would take 9 hours for the timer to complete.
3. Press PB2 and observe the timer.
4. Rung 0001 will have moved a new value of 1000 into the *Preset* of the timer.
5. This is still going to require you to wait the remainder of the 1000 seconds (16 minutes) for the timer to complete.
6. Press PB3 and observe the timer and the green light.
7. Pressing PB3 “energized” rung 0002 and the value previously moved into the *Preset* was moved into the accumulation (*Accum*) word.
8. By definition, when *Accum* = *Preset*, the timers done (DN) bit shall be set on the timer variable T4:0/DN. Since the done bit is true, the green pilot light illuminated.
9. One issue with this program is that the moved value is stored in two places, the move function in rung 0002 and the move function in 0003. If it was determined that 1100 seconds was the new time, it would require a change in two places and for those values to equal. We can simplify the program to ensure that the value need only be changed at one location.
10. Modify the program to match the logic found below.



1. Retest the program to ensure is behaves as it did before the modification. You can press PB1 to reset the timer at any time.
2. Now, alter the program so that 1100 seconds is moved into the timer *Preset* value.
3. Download and test the program. It should behave as before yet we only needed to change the value in one instruction versus two.
4. Moves can also be used to set a number of bits all at the same time.
5. Program the logic found below.



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. Move the two-position selector switch to the “A” (up) position. 2. This will give rung 0000 rung continuity and a integer 9 will be moved into the output word O:0.0 (ours output channels). 3. Both the green and blue pilot lights illuminate since 9 equals 1001 binary. 4. That binary pattern correlates to blue (1), red (0), yellow (0) and green (1). 5. Move the two-position selector switch to the “B” (down) position. 6. Both the yellow and red pilot lights illuminate since 9 equals 0110 binary. 7. That binary pattern correlates to blue (0), red (1), yellow (1) and green (0). | |  |  |  |  | | --- | --- | --- | --- | | 3 | 2 | 1 | 0 | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 1 | 1 | | 0 | 0 | 1 | 0 | 2 | | 0 | 0 | 1 | 1 | 3 | | 0 | 1 | 0 | 0 | 4 | | 0 | 1 | 0 | 1 | 5 | | 0 | 1 | 1 | 0 | 6 | | 0 | 1 | 1 | 1 | 7 | | 1 | 0 | 0 | 0 | 8 | | 1 | 0 | 0 | 1 | 9 | | 1 | 0 | 1 | 0 | 10 | | 1 | 0 | 1 | 1 | 11 | | 1 | 1 | 0 | 0 | 12 | | 1 | 1 | 0 | 1 | 13 | | 1 | 1 | 1 | 0 | 14 | | 1 | 1 | 1 | 1 | 15 | |

1. Experiment with the program changing the numbers from anywhere between 0 and 15.

This page left intentionally almost blank