

This program is totally useless...

as far as controlling any process. Now on the other hand if you would like to see how different instructions are converted read on.

This program was converted using the DHRIO option. Bits in the rack range will be remapped to I/O. Bits outside of this range will point to the INT array (I000 or O000).

I/O bit addresses use the format: `_RRS:I.Data[M].B`
 where: RR = rack in octal
 S = Starting module group
 I = Input (or O for output)
 M = Module group offset
 B = Terminal number in decimal

0

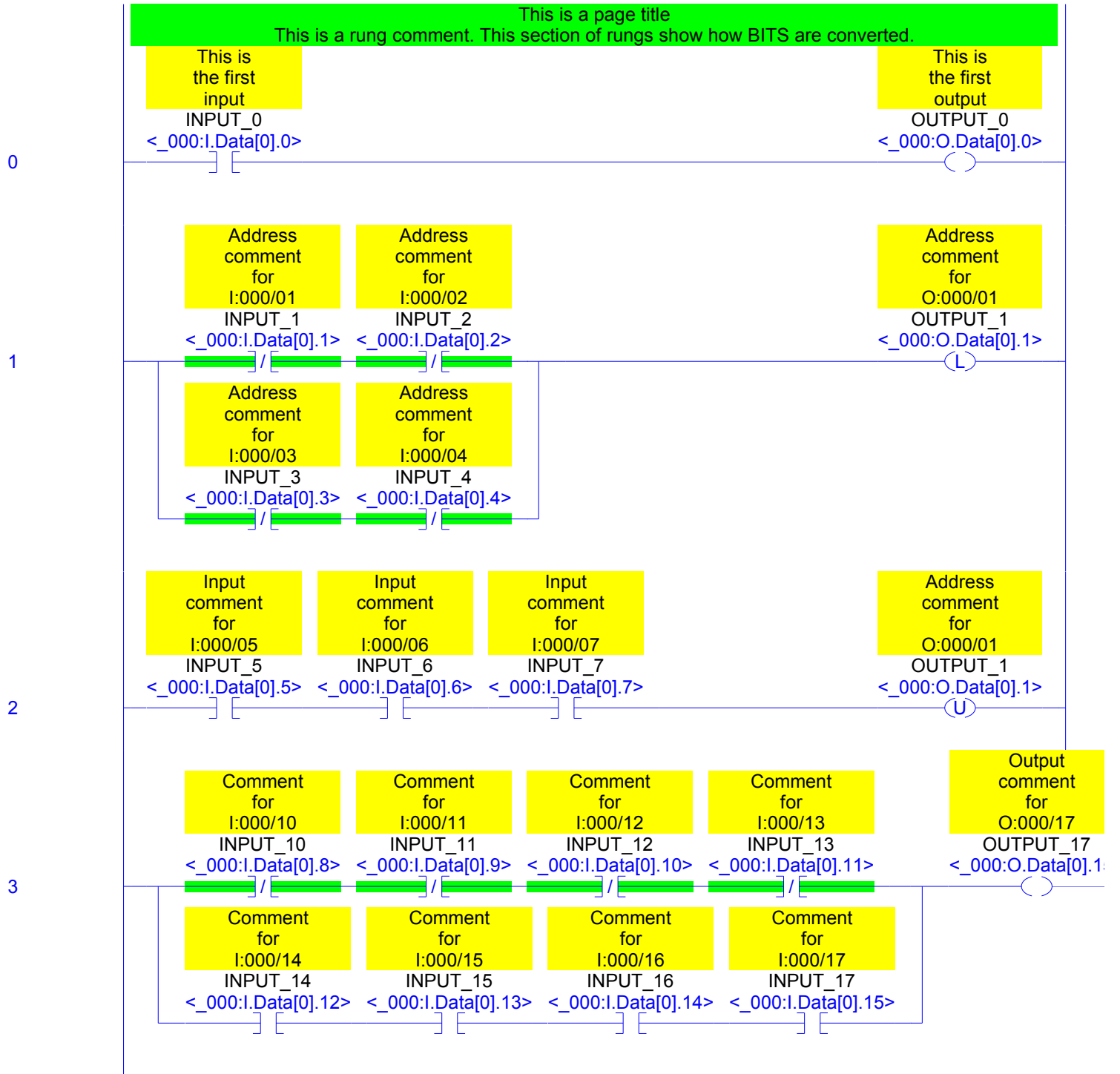
JSR
 Jump To Subroutine
 Routine Name LAD_003_BITS

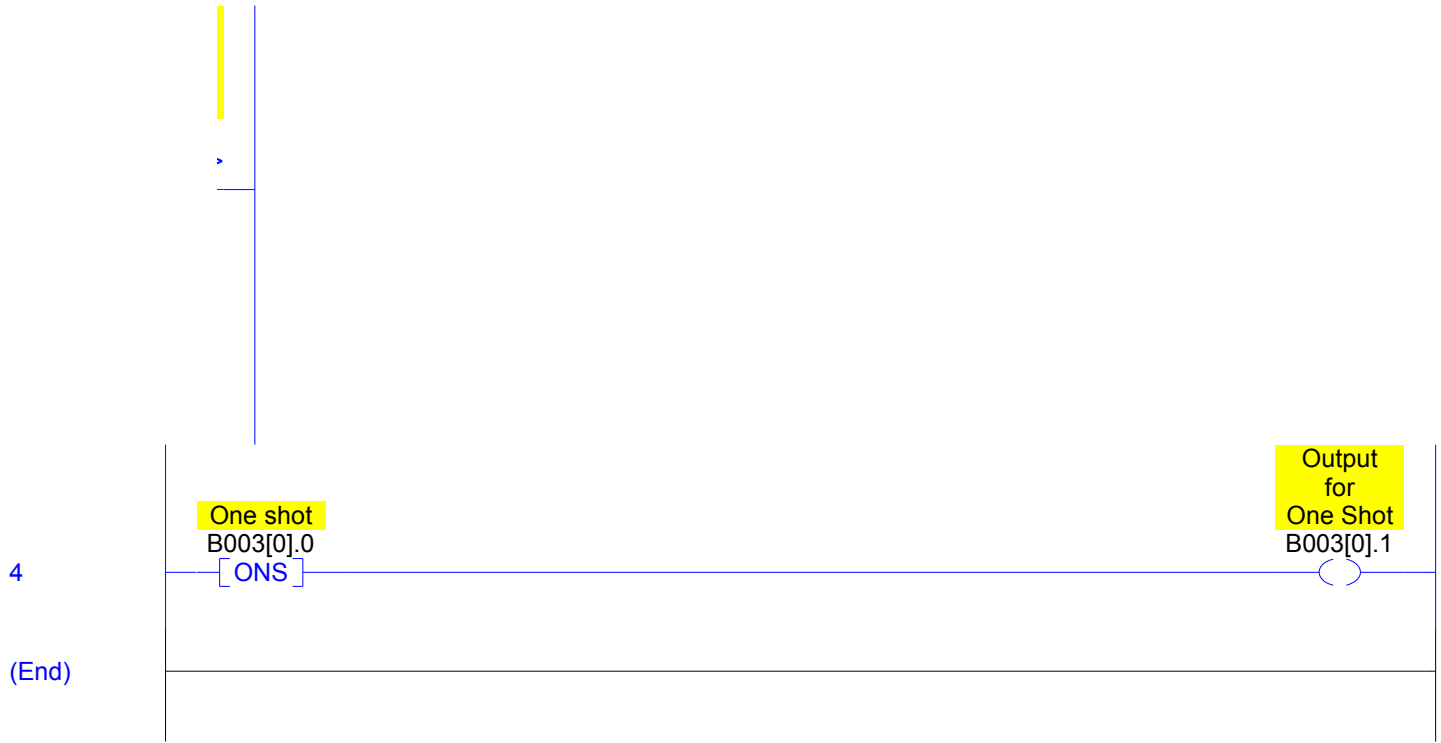
1

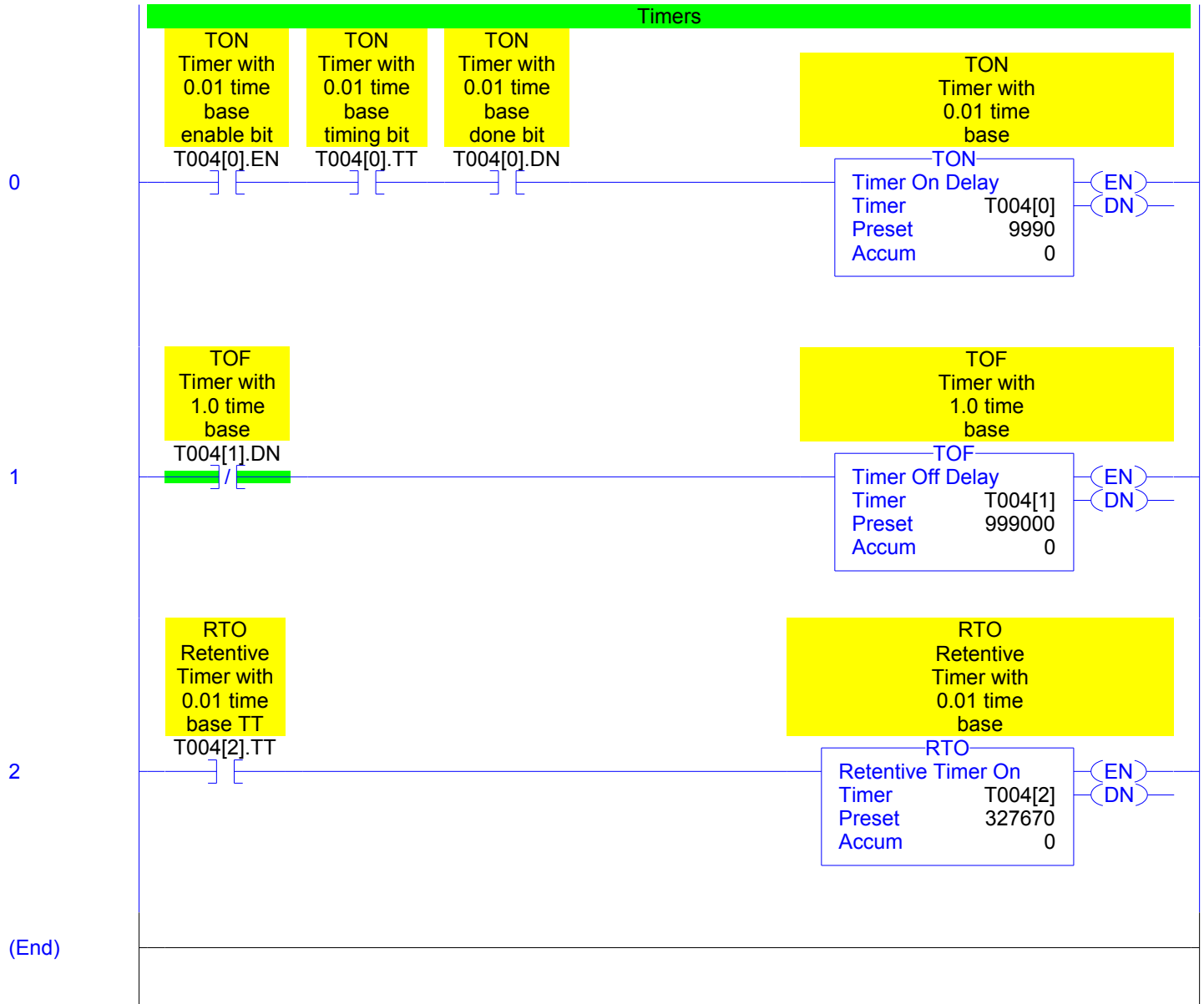
Real time
 clock YEAR

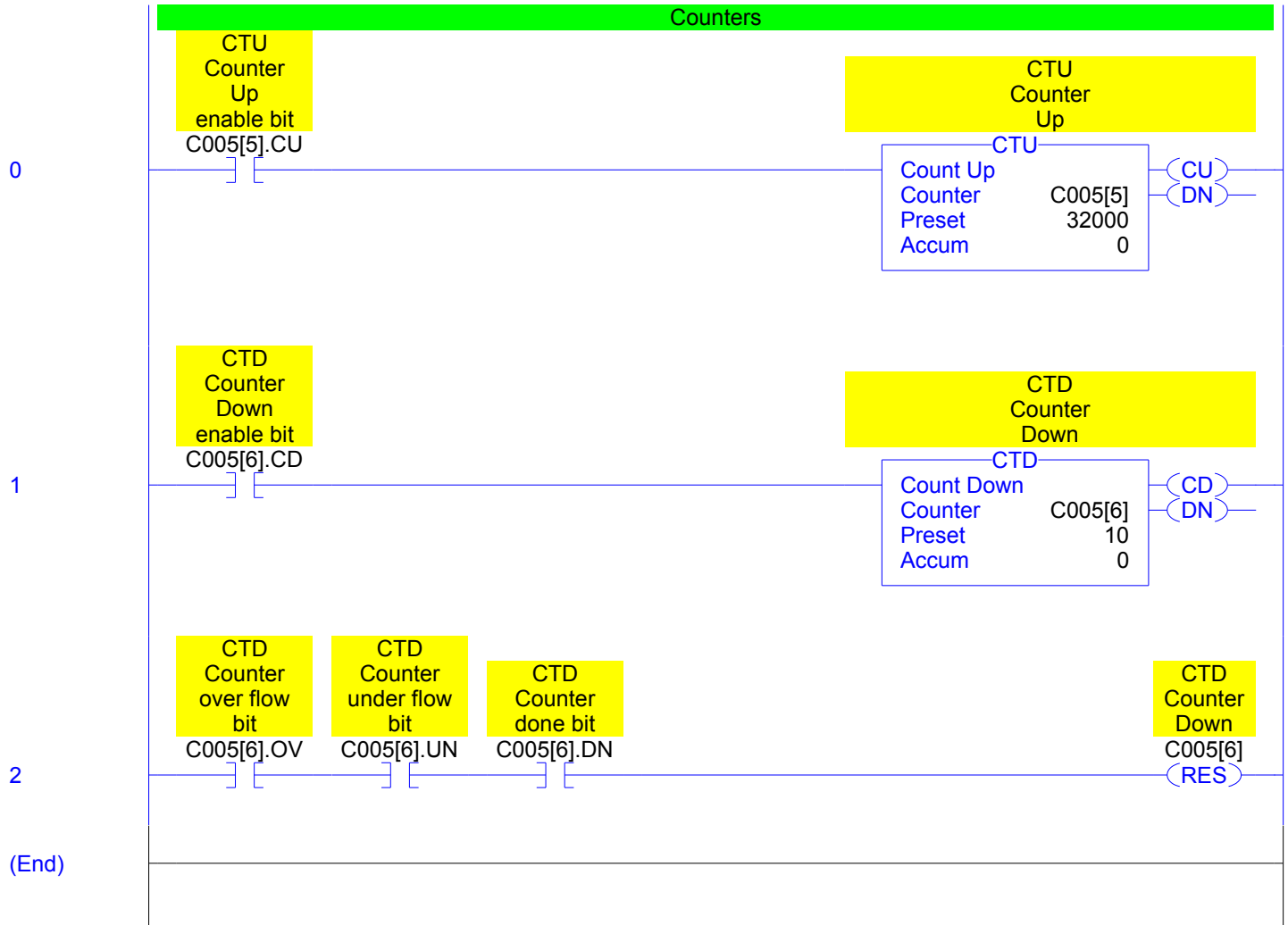
GSV
 Get System Value
 Class Name WALLCLOCKTIME
 Instance Name
 Attribute Name DateTime
 Dest S000[18]
 0

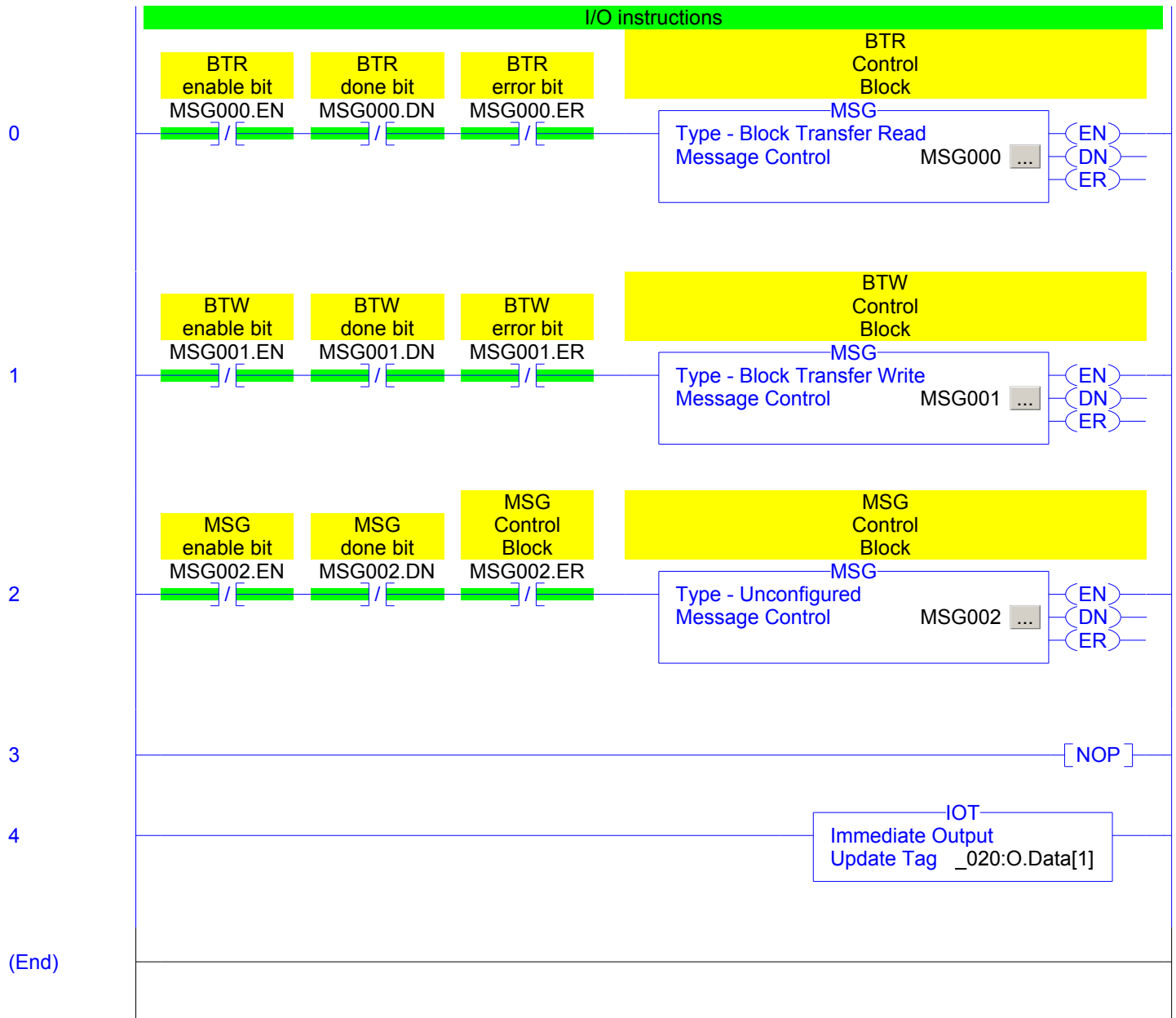
(End)



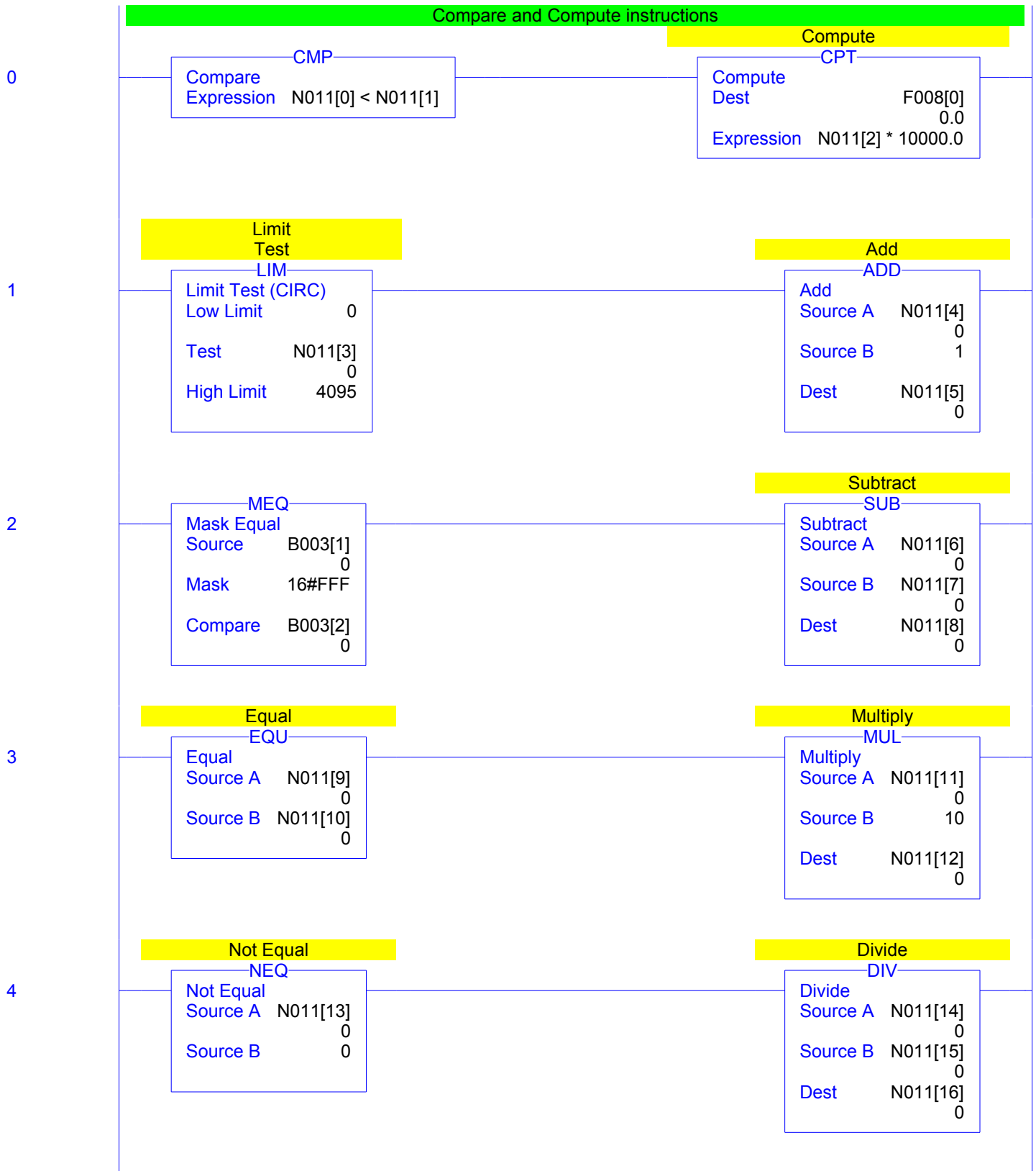


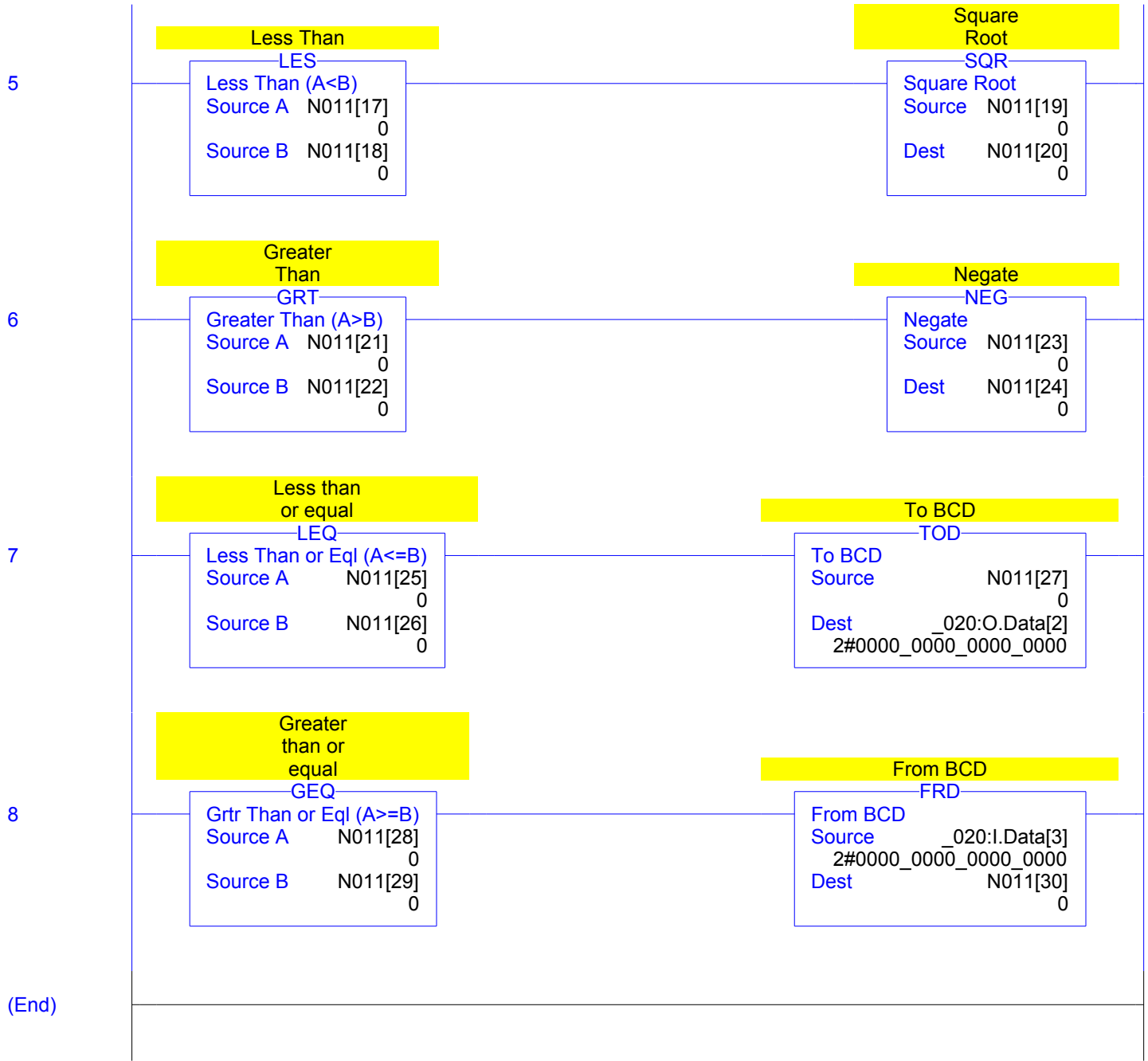


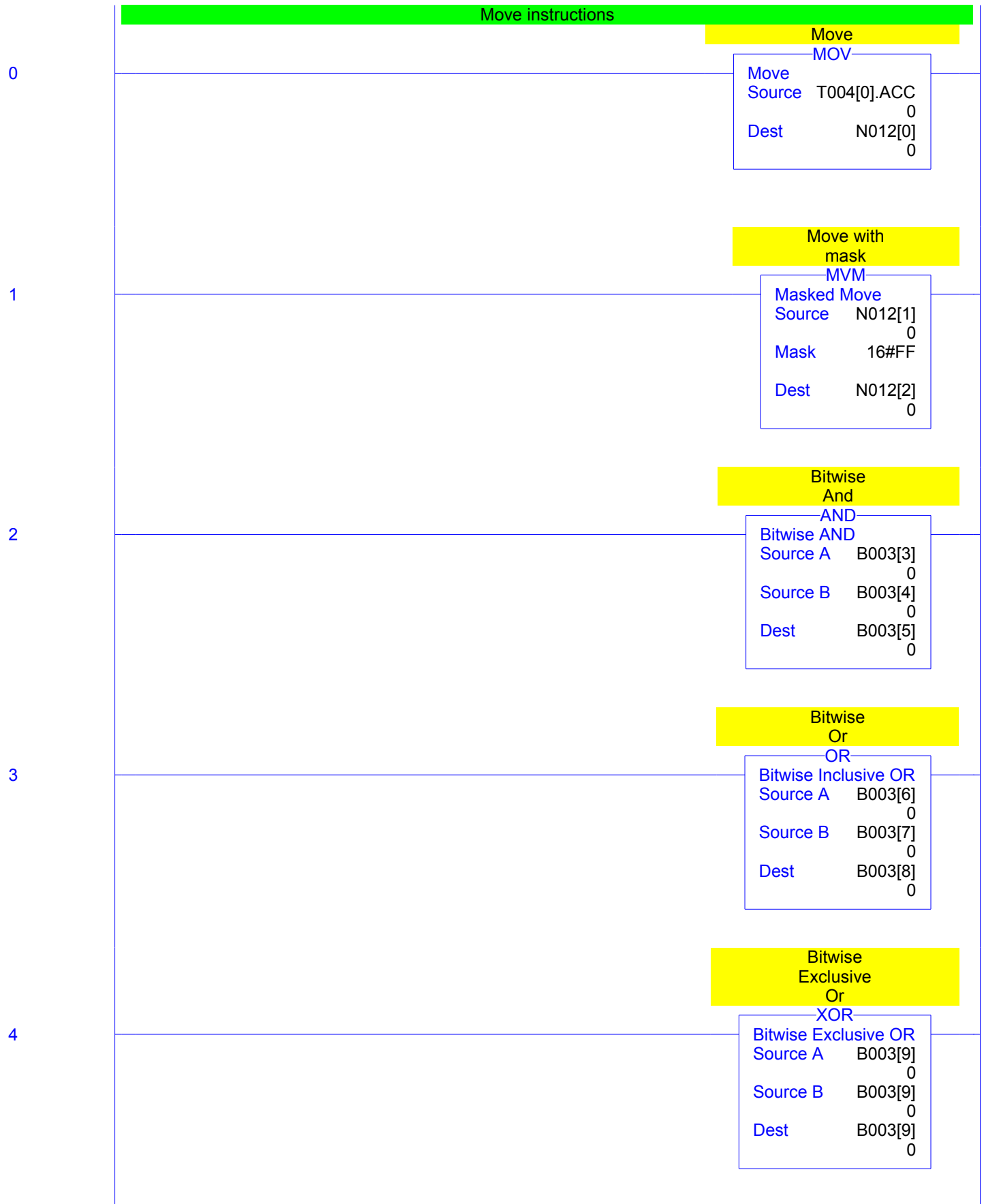


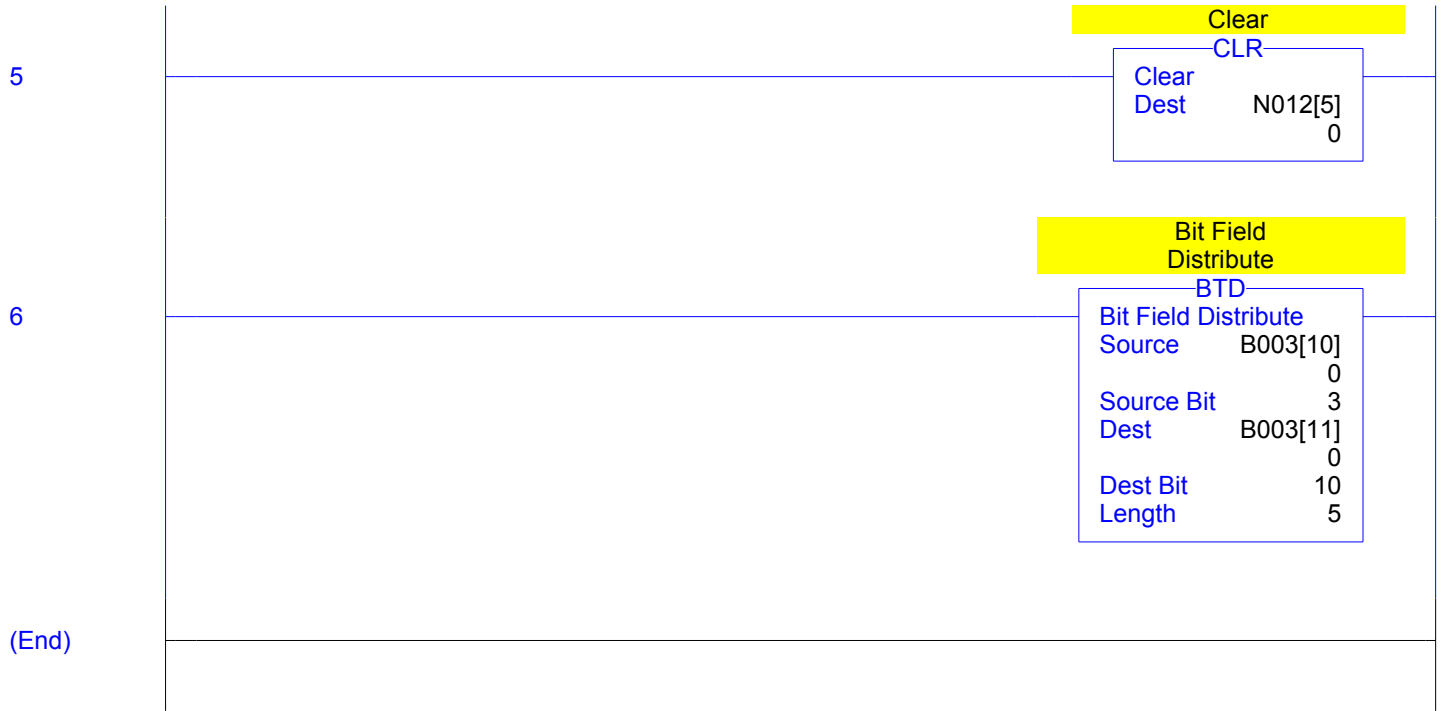


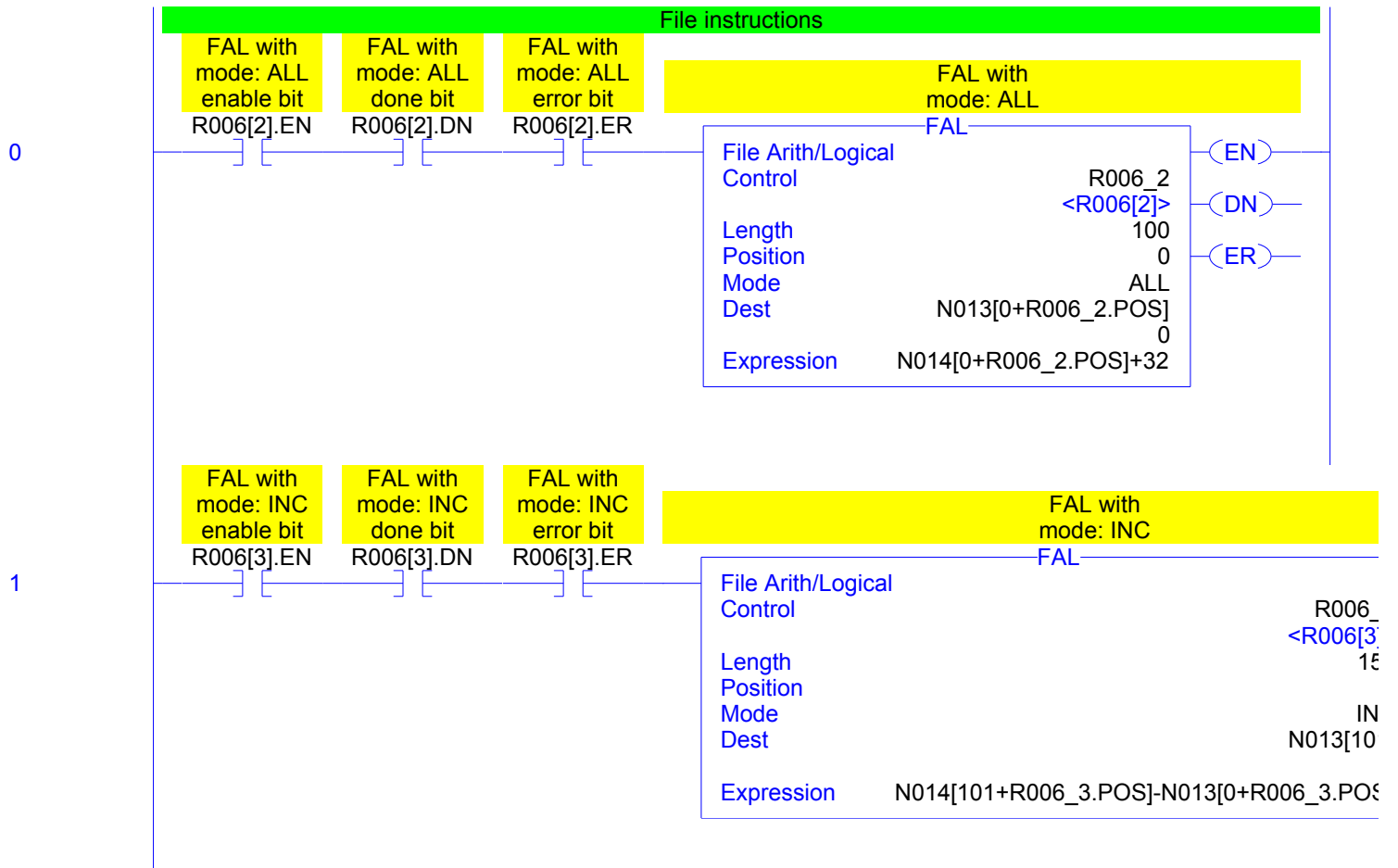
Compare and Compute instructions

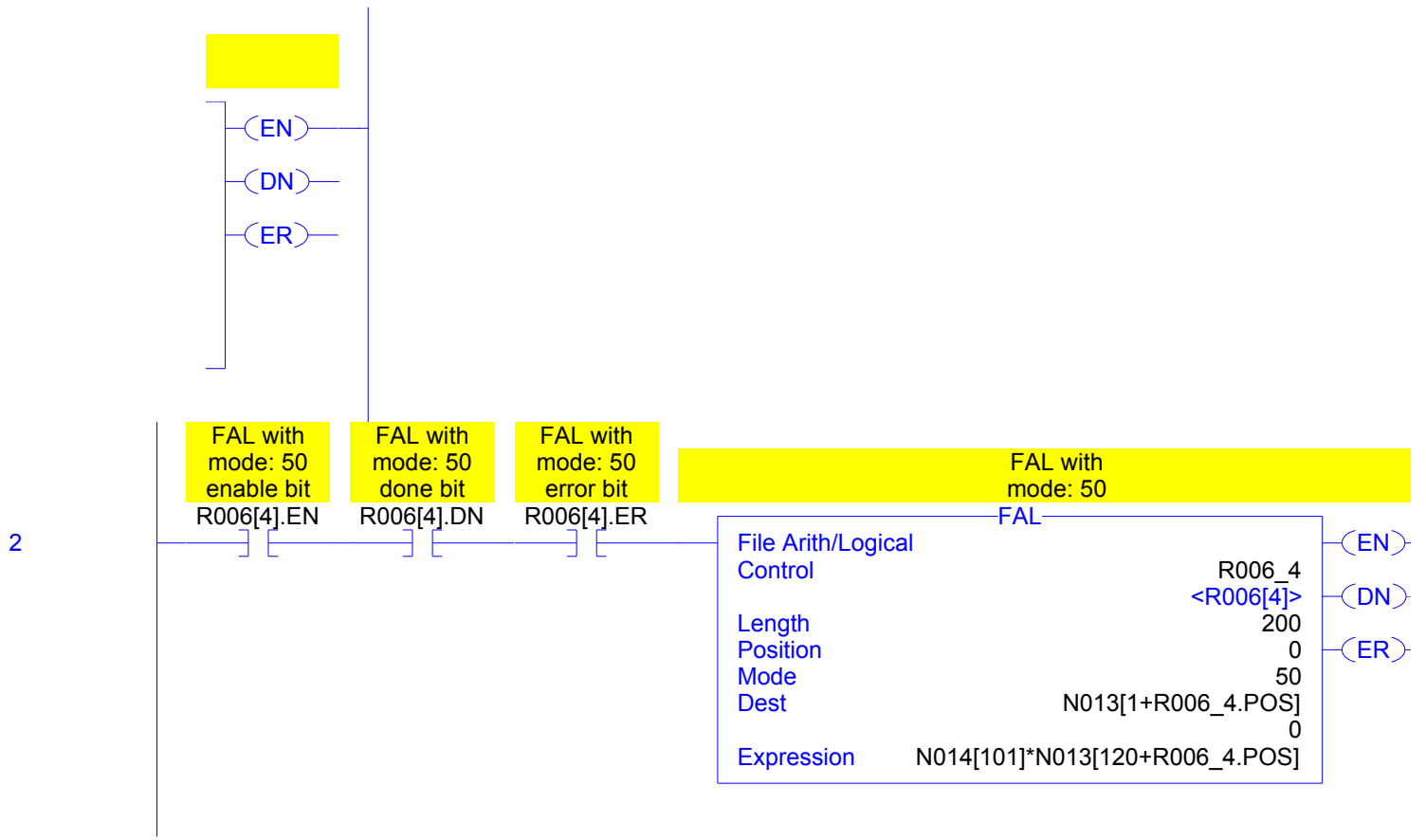


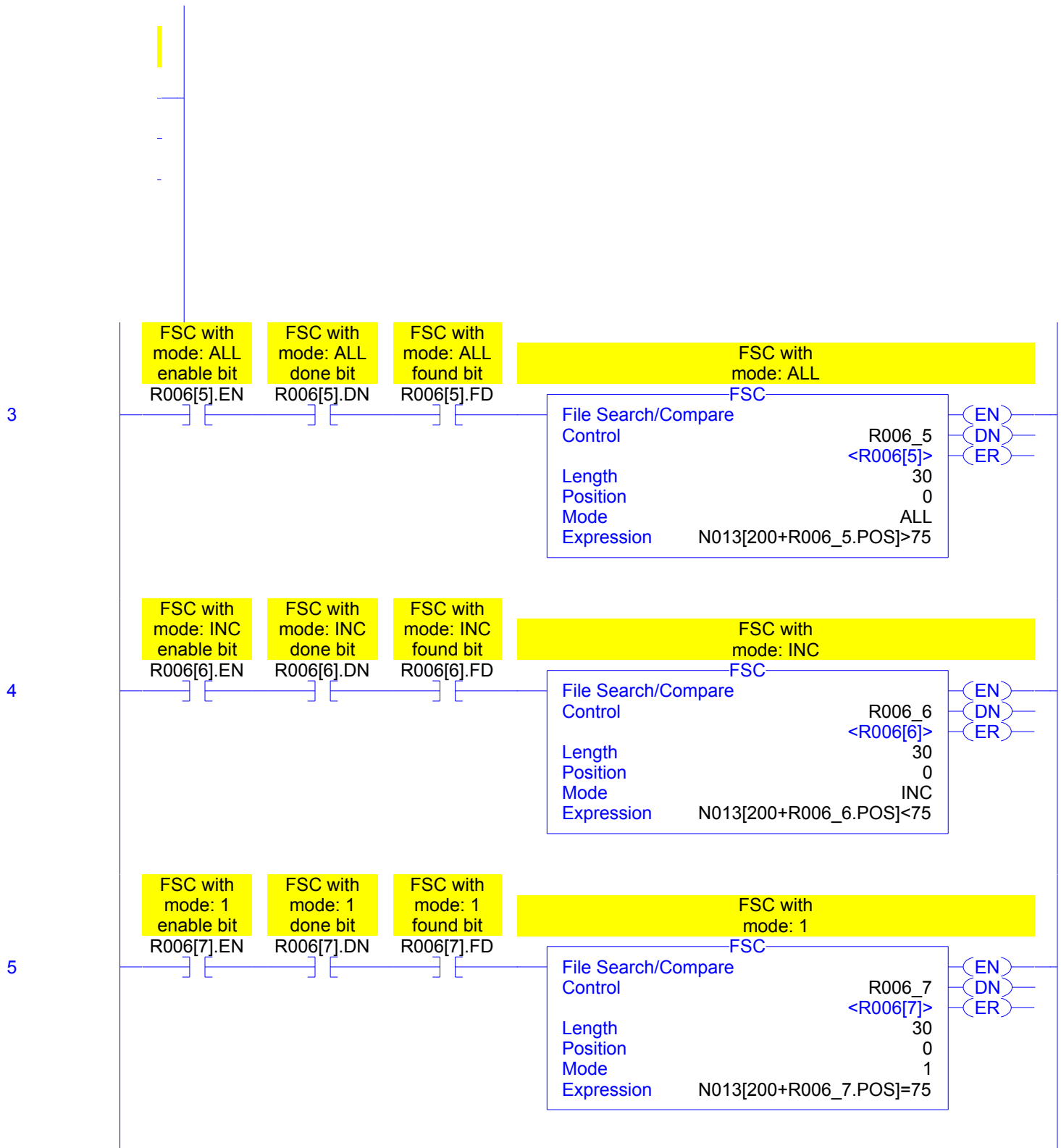


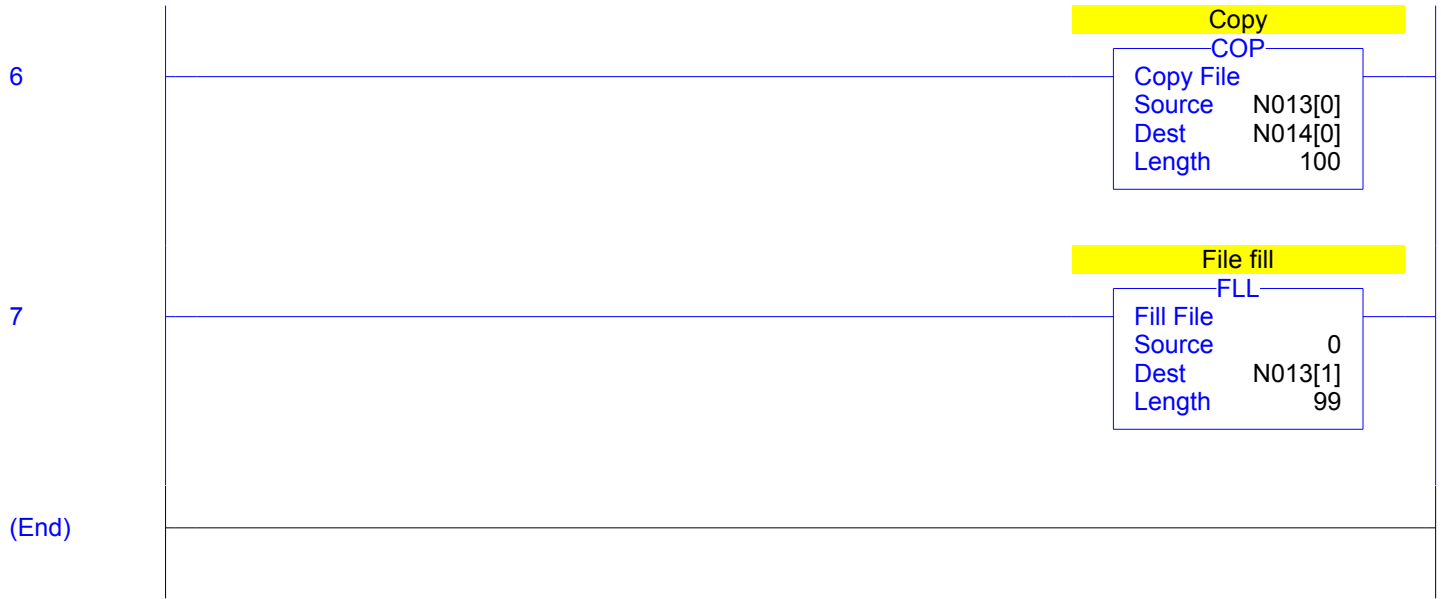


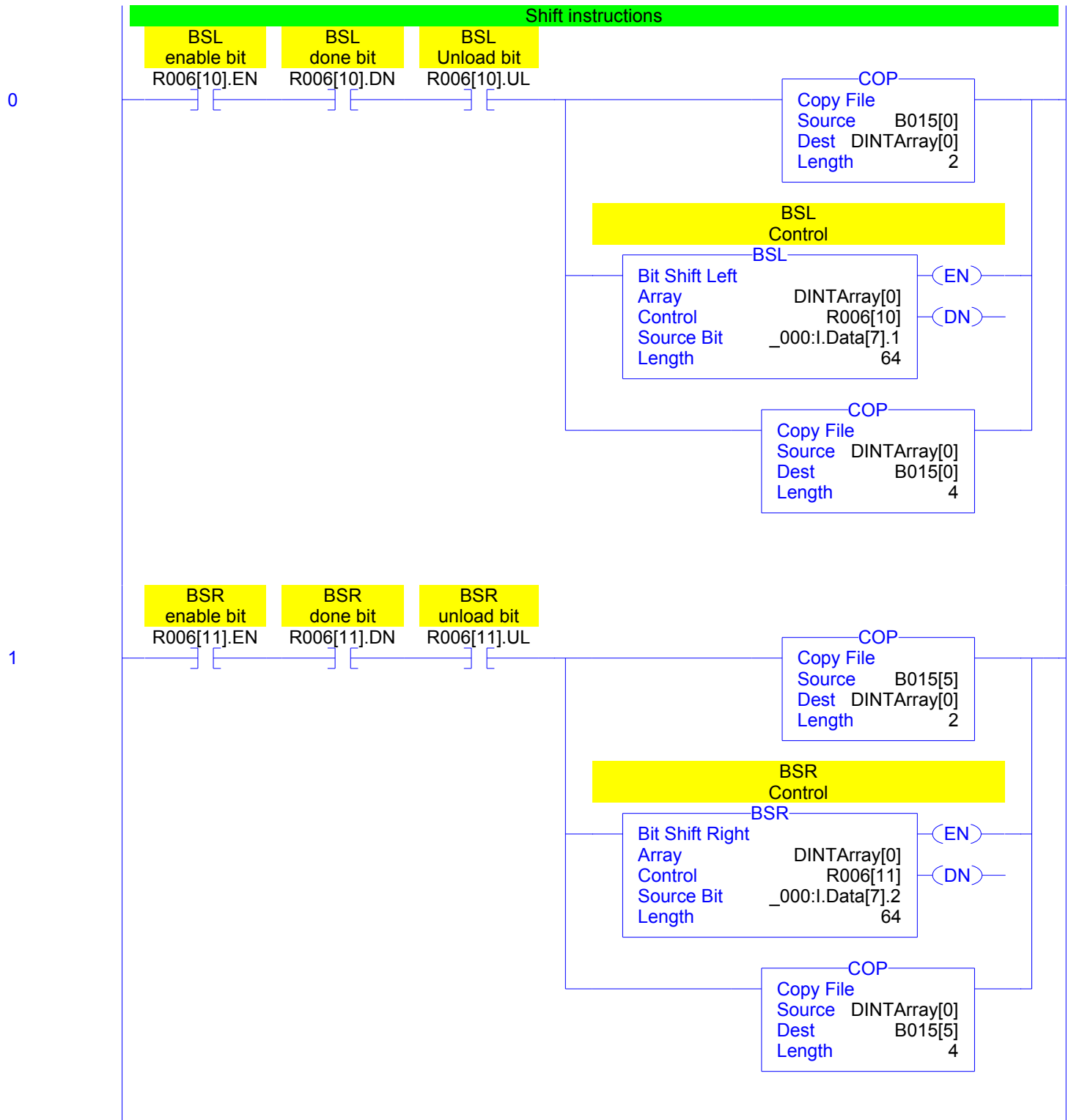


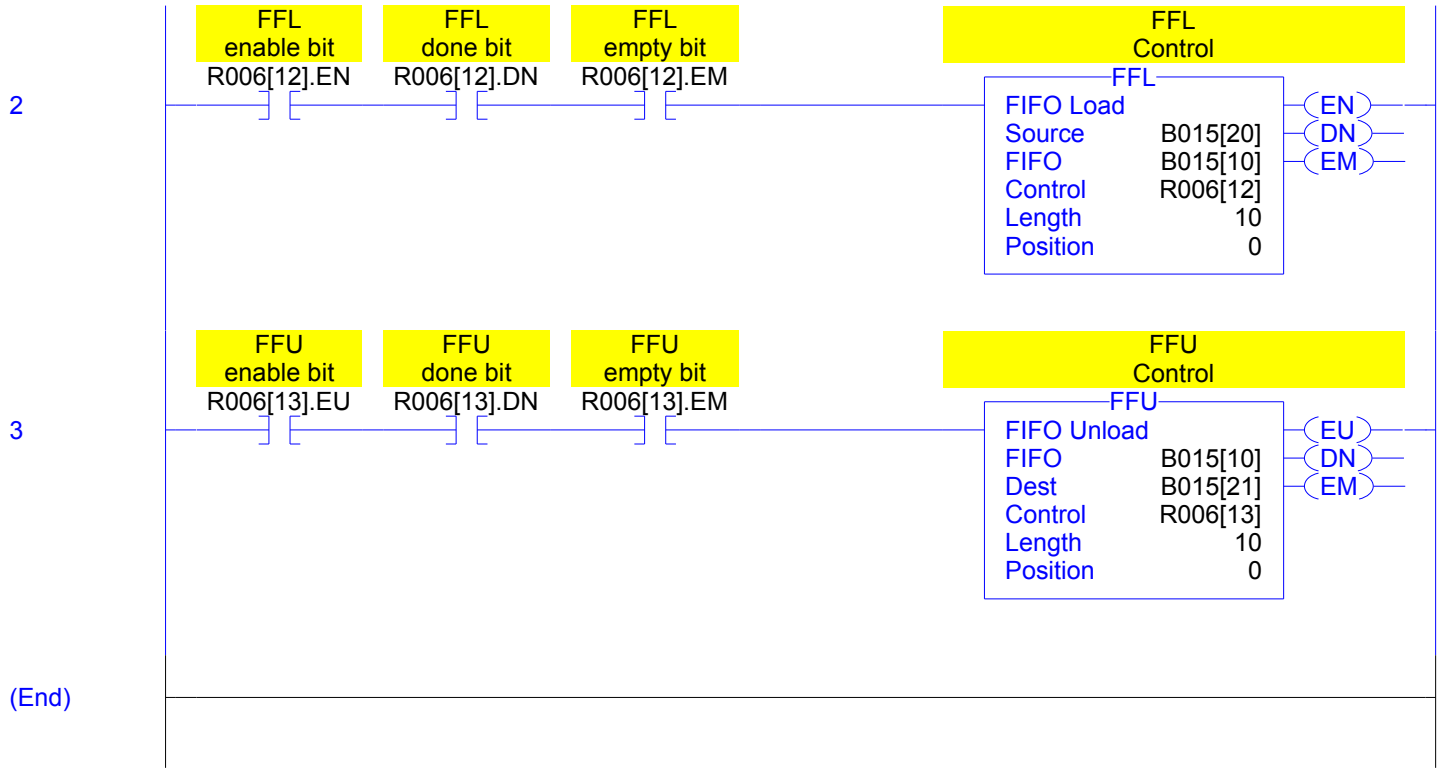


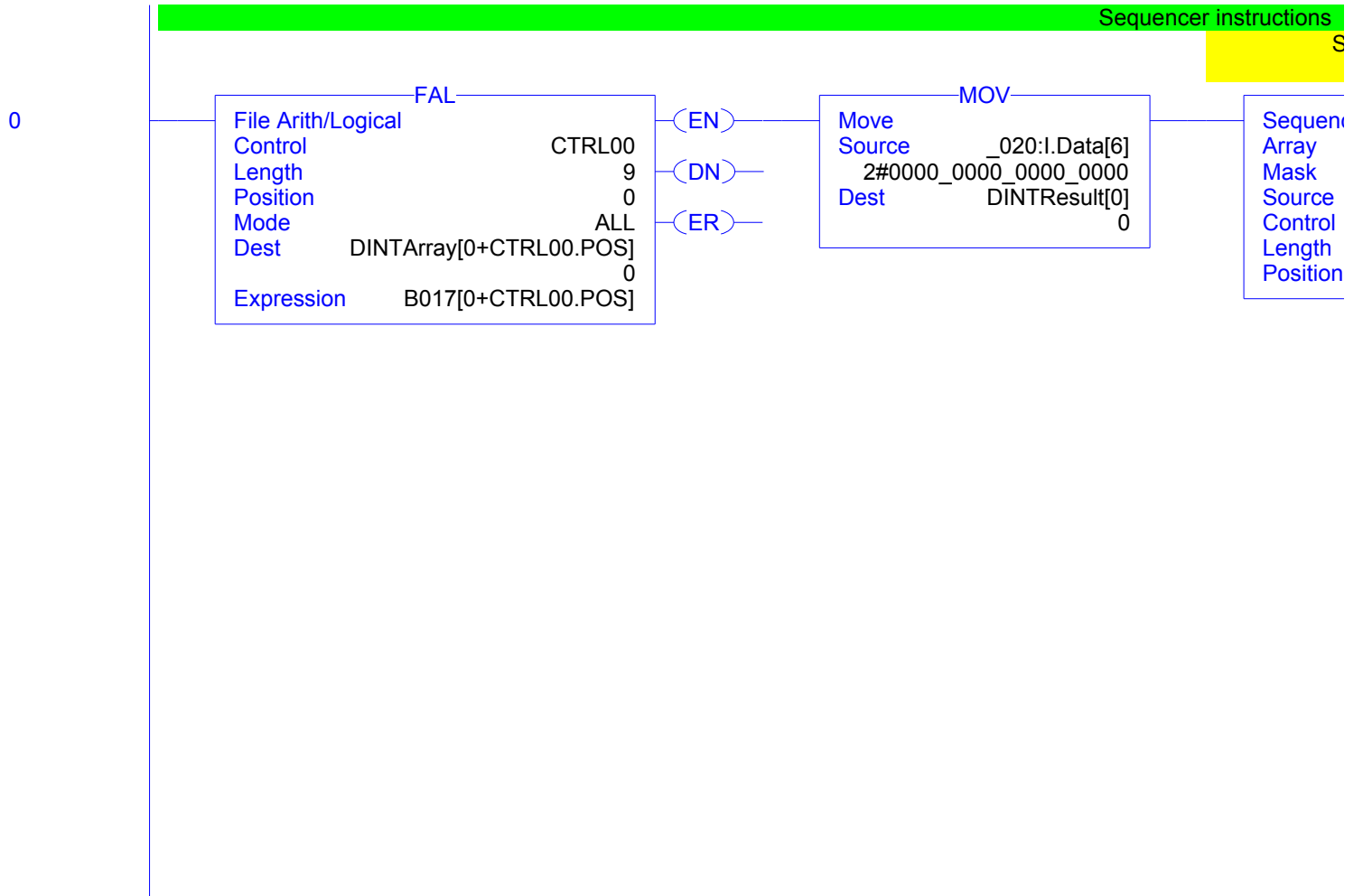


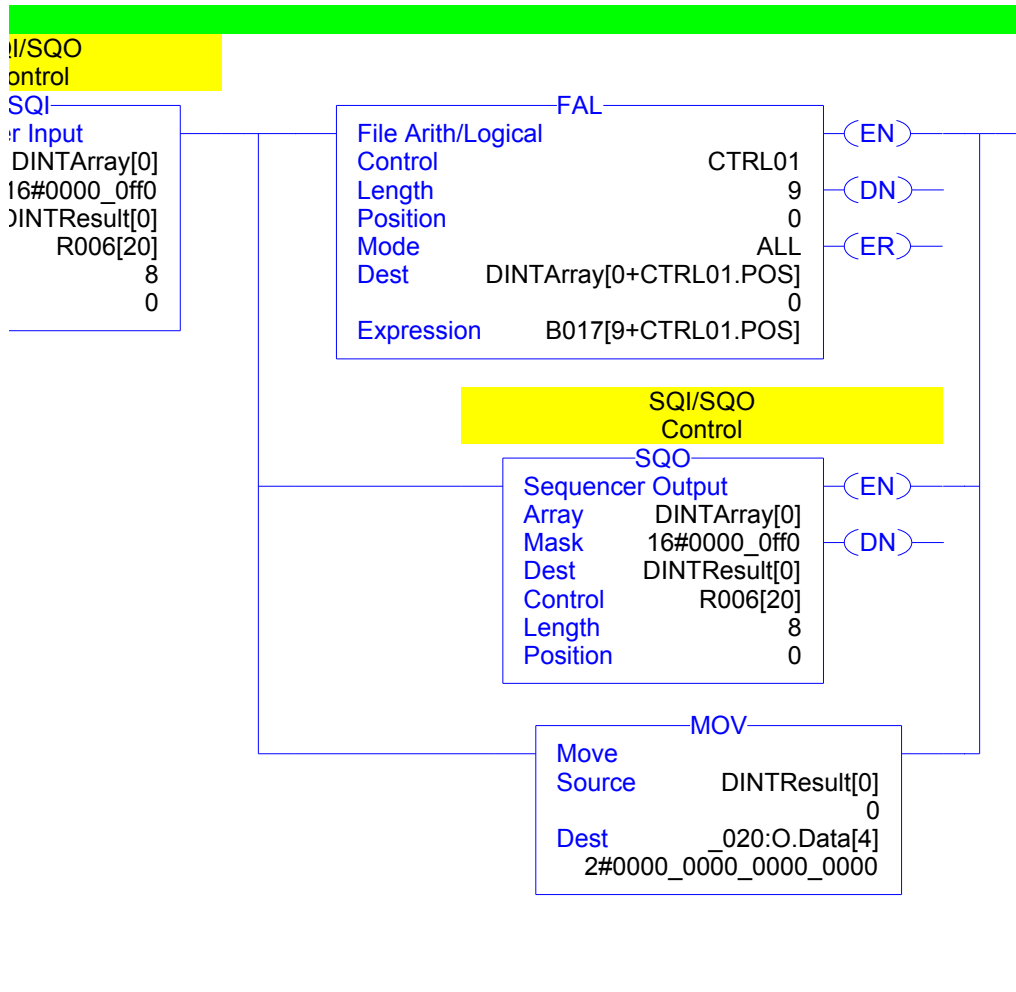












1

Load
sequence
file
input
condition

_000:I.Data[1].8

FAL

File Arith/Logical	
Control	CTRL02
Length	9
Position	0
Mode	ALL
Dest	DINTArray[0+CTRL02.POS]
	0
Expression	B017[0+CTRL02.POS]

(EN)
(DN)
(ER)

MOV

Move	
Source	_020:I.Data[6]
	2#0000_0000_0000_0000
Dest	DINTResult[0]
	0

SQL
Control

SQL

Sequencer Load	
Array	DINTArray[0]
Source	DINTResult[0]
Control	R006[21]
Length	8
Position	0

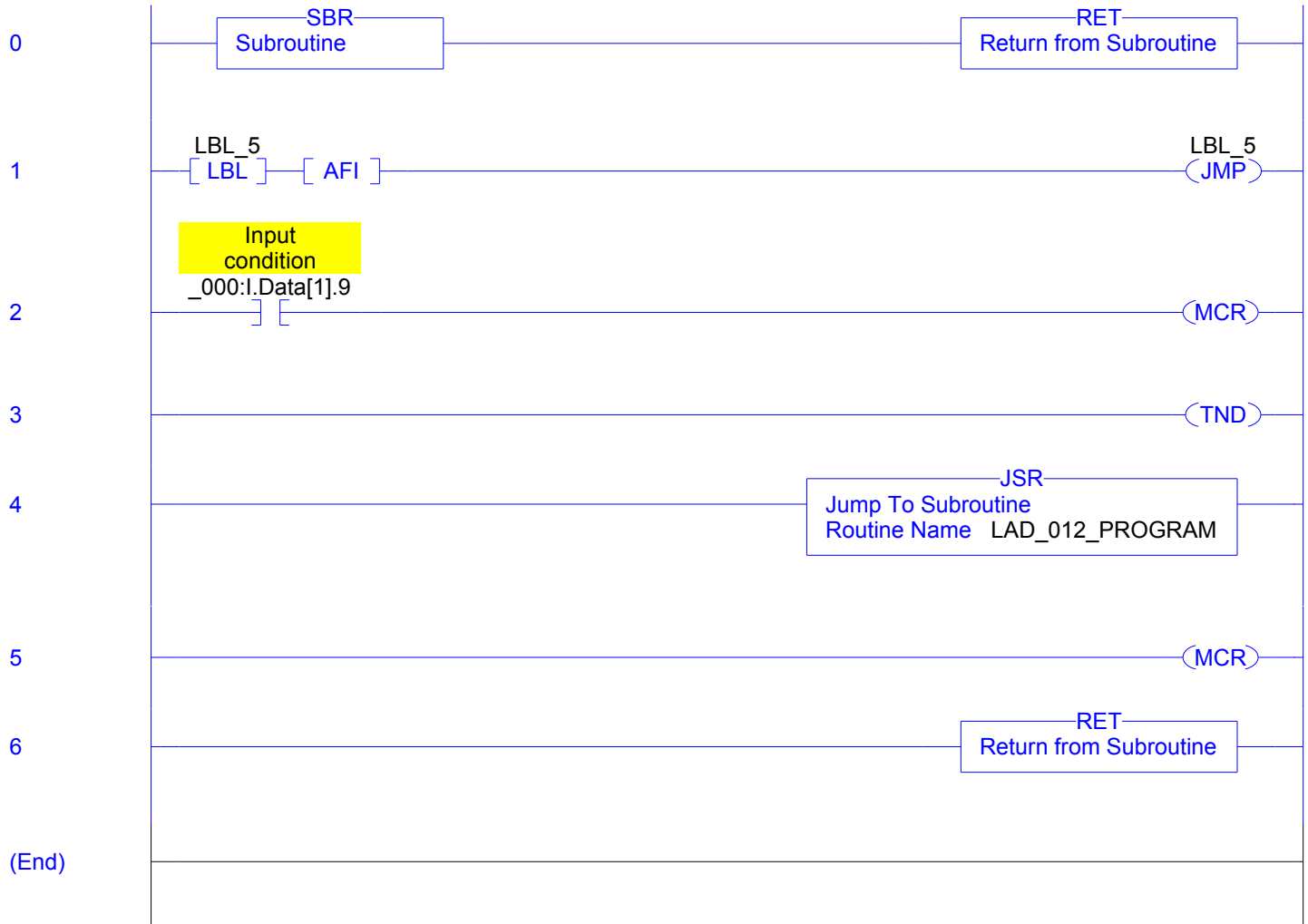
(EN)
(DN)

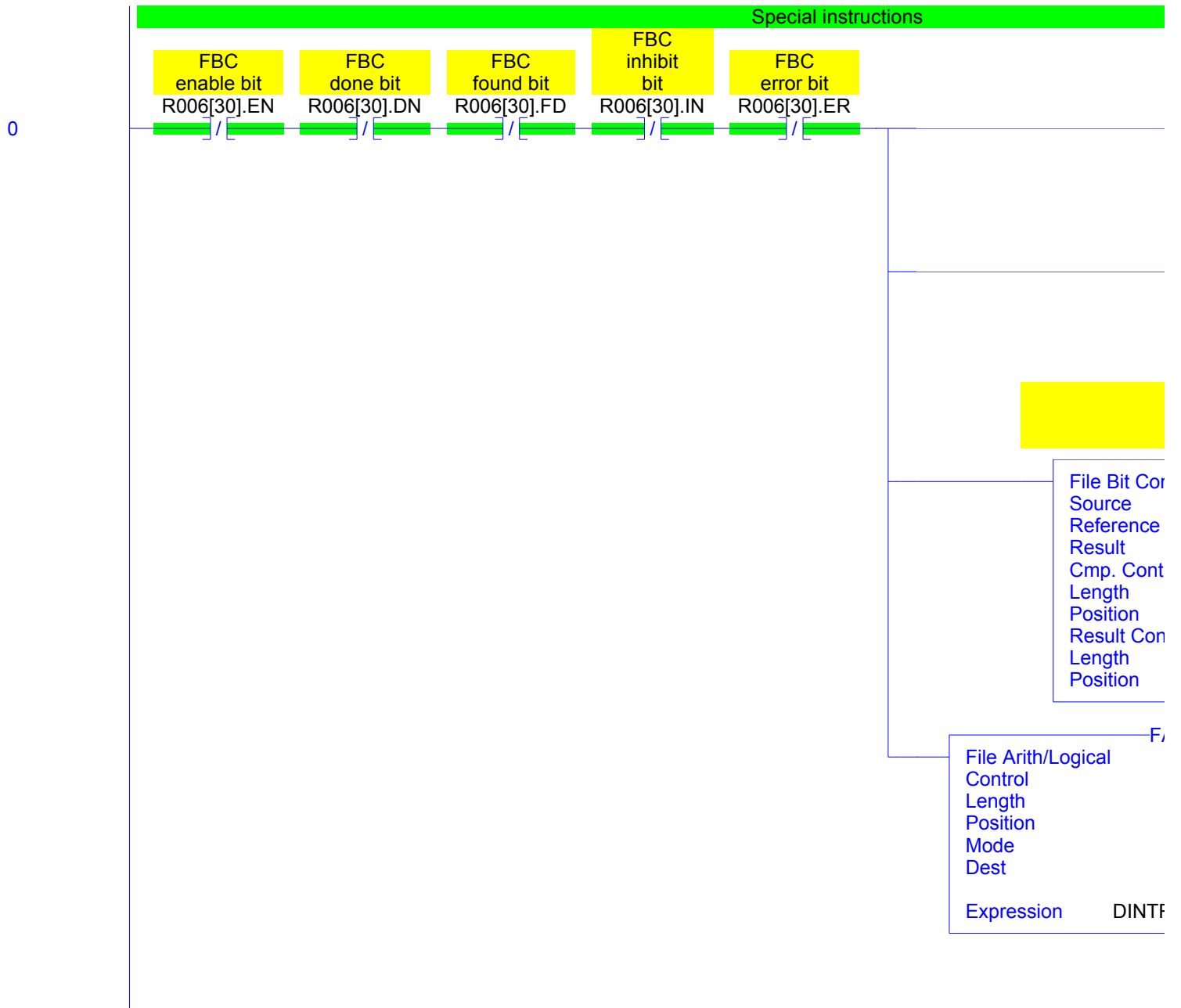
FAL

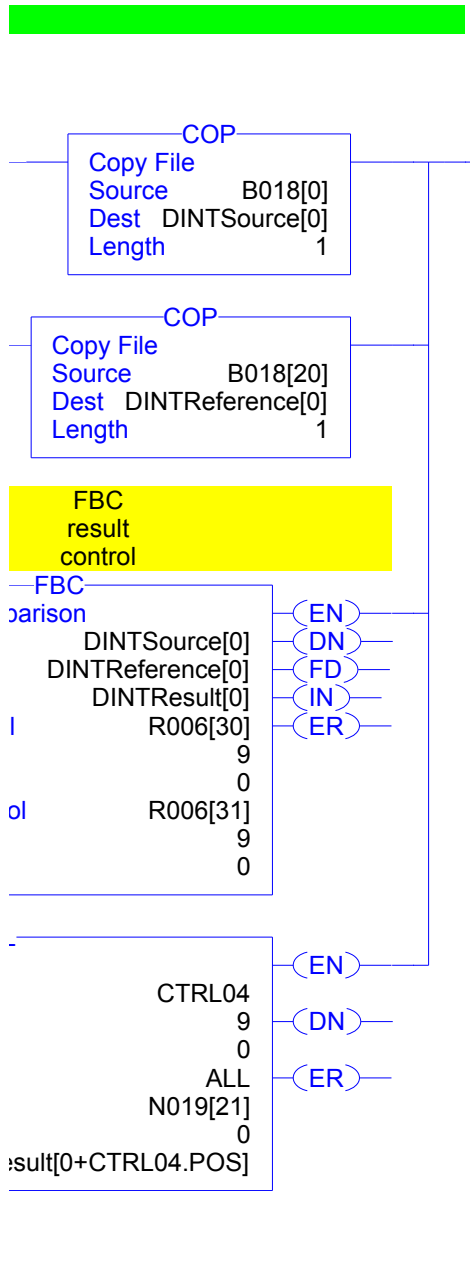
File Arith/Logical	
Control	CTRL03
Length	9
Position	0
Mode	ALL
Dest	B017[0+CTRL03.POS]
	0
Expression	DINTArray[0+CTRL03.POS]

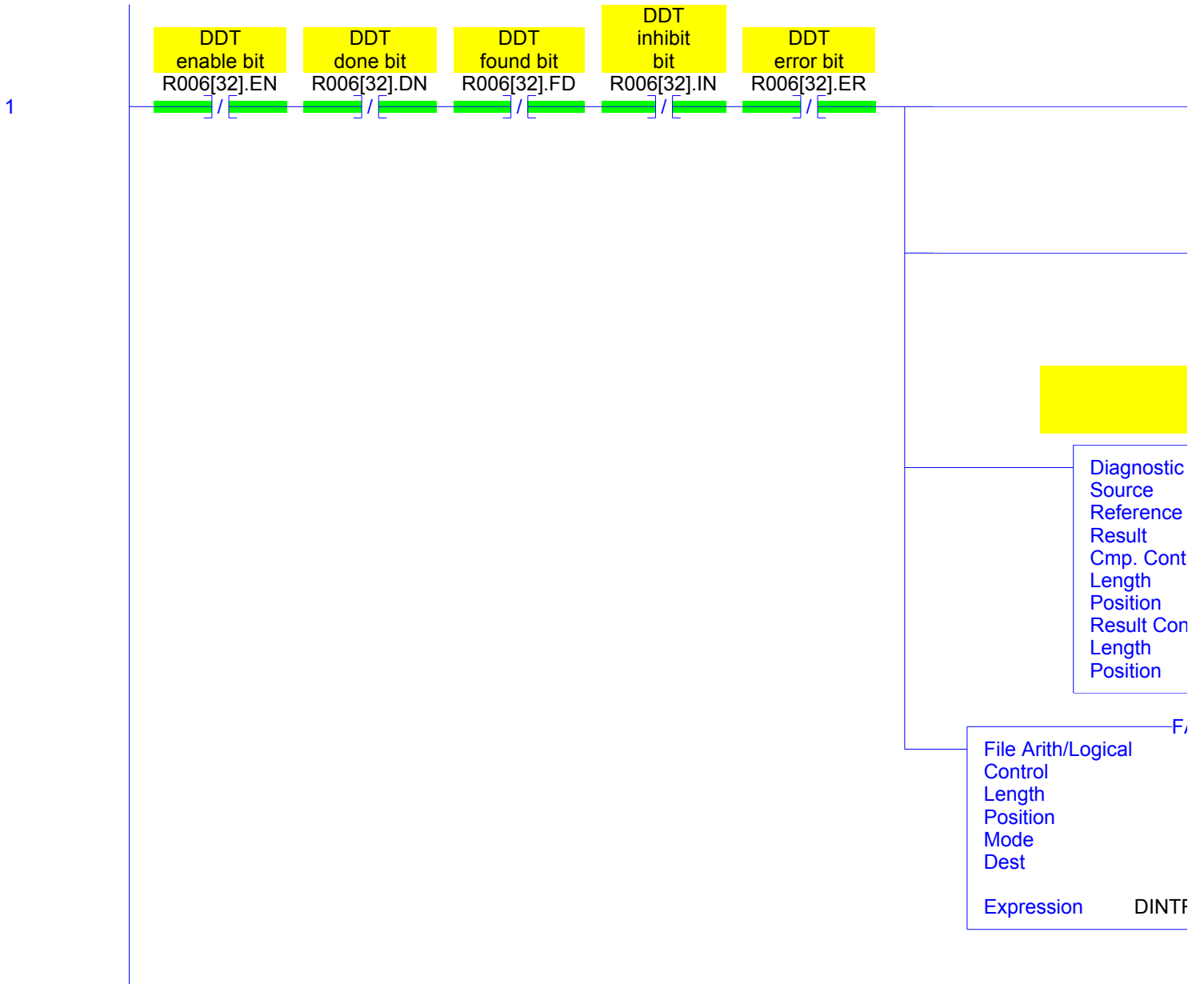
(EN)
(DN)
(ER)

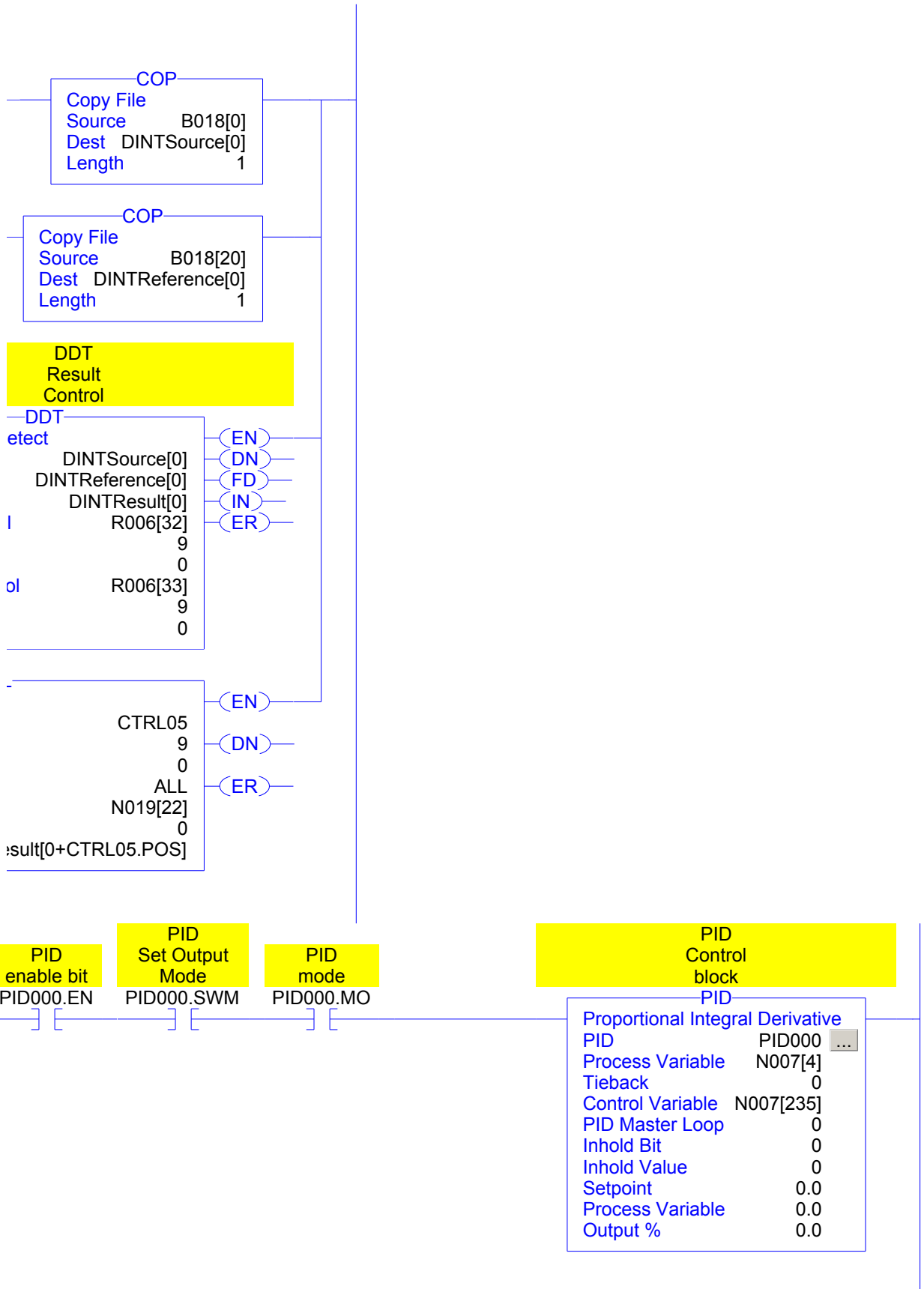
(End)











(End)

