1

```
Simple quadrature decoder. Gary Dickinson 2/14/2020

This is a very simplified version of Lorne Van Dusen's code.
I just got rid of the stuff that didn't add value.

This version uses ladder logic COUNTERs. I agree with Lorne's
observation that the ladder logic COUNTERs are a bit obtuse.
A single COUNTER can work with values between 0 and 9999.
It is possible to extend the count range by cascading COUNTERs.
I used 2 counters to get a count range 0 to 99,999,999.
```

2

```
1st.Scan                                                    LoadCntrs
 | |                                                       2{dCusF}
```

3

```
Reset                                                       ZeroCntrs
 | |r3                                                     1{dCusF}
```

4

```
The following two rungs detect the direction based on the
state of the A input at the rising edge of the B input.
CntUP or CntDN will be active for one scan time.
```

5

```
B          A                                                CntUP
 |↑|r2     | |r1                                          r4(RLY)
```

6

```
B          A                                                CntDN
 |↑|r2     |/|r1                                          r5(RLY)
```

7

```
The following 2 rungs increment a cascaded pair of "up" COUNTERs.
Each PLC COUNTER can counters can count from 0..9999.
When the first counter overflows from 9999 to 0 the 2nd COUTNER
is incremented.
```

8

```
CntUP                                                       Digits3_0
 | |r4                                                     c9[Upctr]
```

9

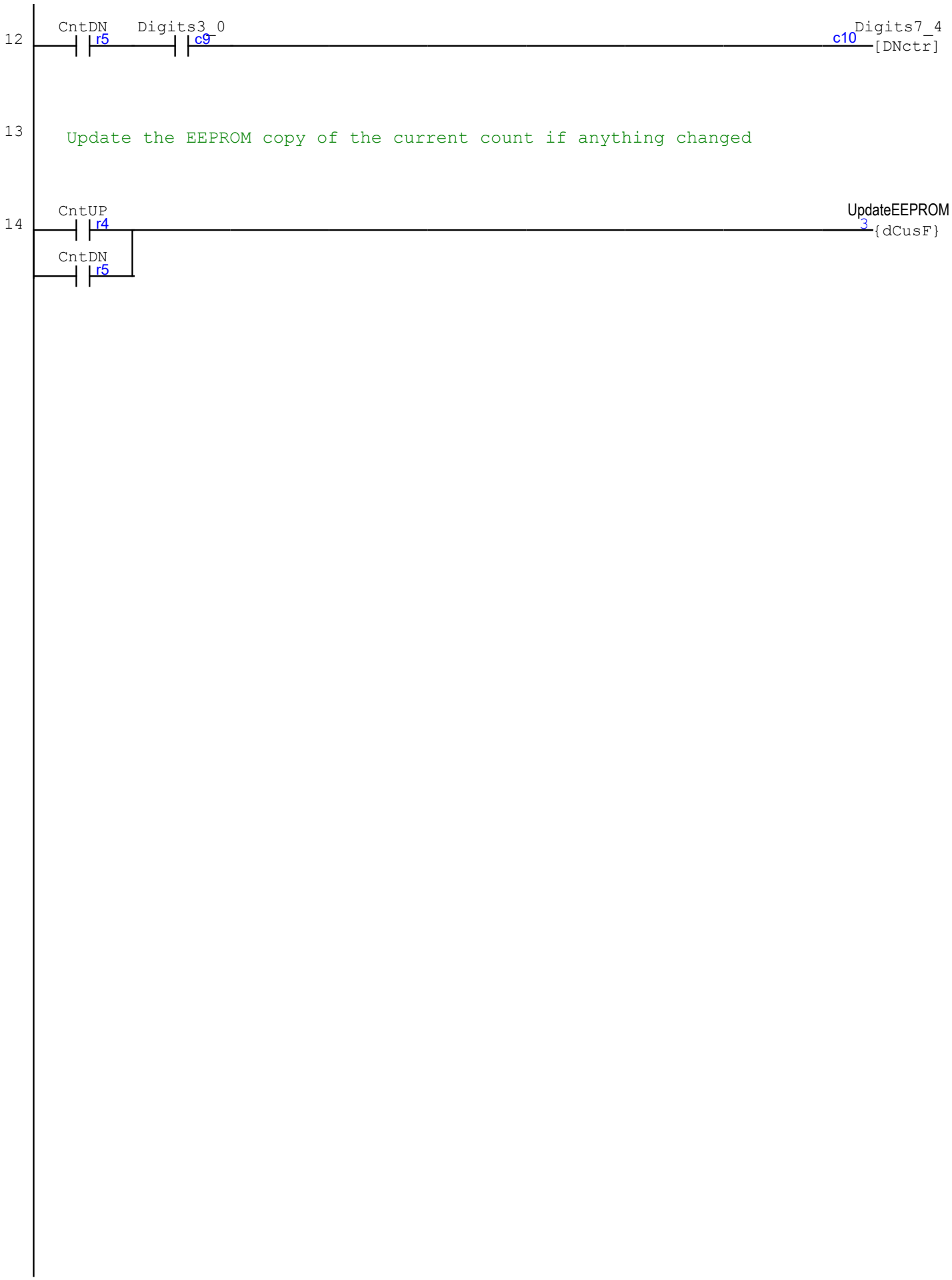
```
CntUP      Digits3_0                                        Digits7_4
 | |r4     |/|c9                                         c10[Upctr]
```

10

```
The following 2 rungs increment a cascaded pair of "down" COUNTERs.
```

11

```
CntDN                                                       Digits3_0
 | |r5                                                     c9[DNctr]
```


12
CntDN
r5
Digits3_0
c9
Digits7_4
c10
[DNctr]
13
Update the EEPROM copy of the current count if anything changed
14
CntUP
r4
CntDN
r5
UpdateEEPROM
3
{dCusF}