1

```
NEVER upload code to the machine remotely, it's possible to physically break
things when this code is altered so you need to be there in person to monitor it
```

2

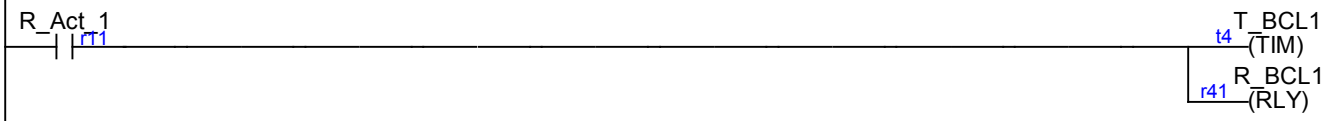
```
================================================================================================
 _____ _
|_   _(_)_ __ ___   ___ _ __ ___
  | | | | '_ ` _ \ / _ \ '__/ __|
  | | | | | | | | |  __/ |  \__ \
  |_| |_|_| |_| |_|\___|_|  |___/

================================================================================================
* Any one of the latching internal relays can turn on the drink dispensing
  timers since we use one set of timers for everything
* We check down the bottom of this file that none of the relays can be on
  at the same time (assertions)
* Dispensing a drink consists of: on 400ms, off 500ms, on again for 660ms
  (as measured before the old electronics were removed from the machine)
* We use combinations of timers to work out when we should start/stop the solenoids
```

3

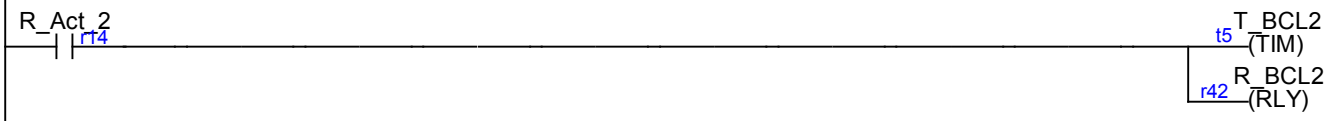
4

```
* Each actuator relay also starts its own "Big Coke Lock" (BCL) counter
  and latch which allows us to check that nothing else is dispensing at
  the same time (see Locks and Checks section)
```

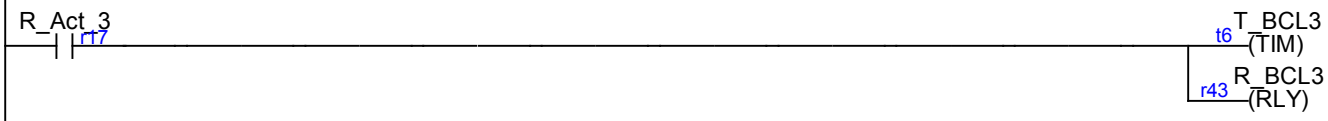
5



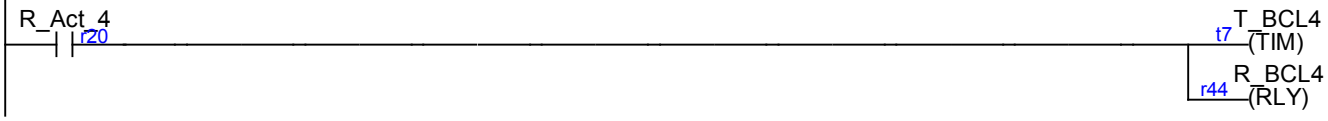
6



7



8



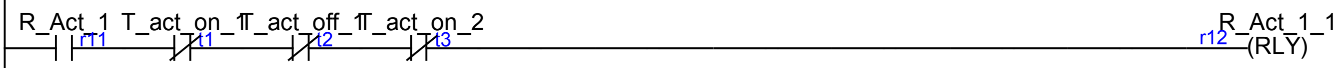
9–14

15

Actuator 1

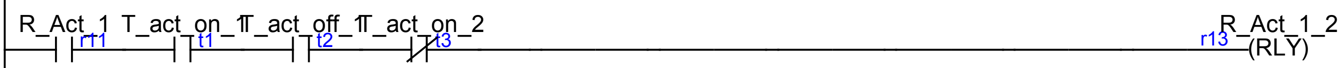
16 * First "on" time

17



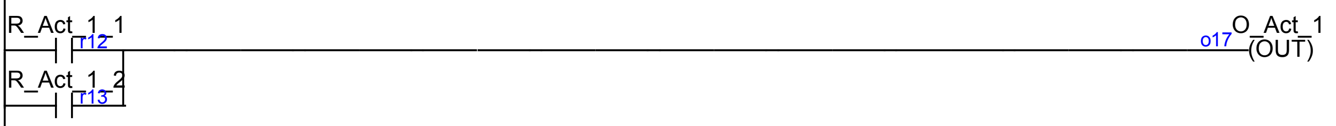
18 * Second "on" time (the off time occurs by logical magic)

19



20 * Turn on physical output when necessary

21

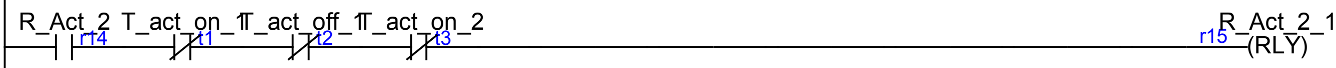


22 * Clear the Dispense1 relay once all the counters are finished, or clear the latch immediately if there's no drink in the slot
* The OR condition here will also fix the solenoid if it gets half a push out of time and the slot empties (drink drops on first push and then registers as empty)

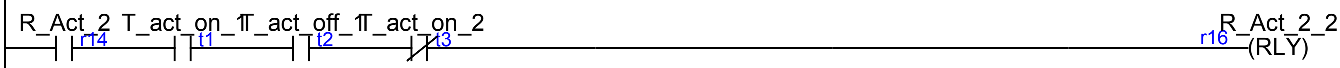
23 

24
```
==========================================================================================
    _        _               _               ____  
   / \   ___| |_ _   _  __ _| |_ ___  _ __  |___ \ 
  / _ \ / __| __| | | |/ _` | __/ _ \| '__|   __) |
 / ___ \ (__| |_| |_| | (_| | || (_) | |     / __/ 
/_/   \_\___|\__|\__,_|\__,_|\__\___/|_|    |_____|

==========================================================================================
```

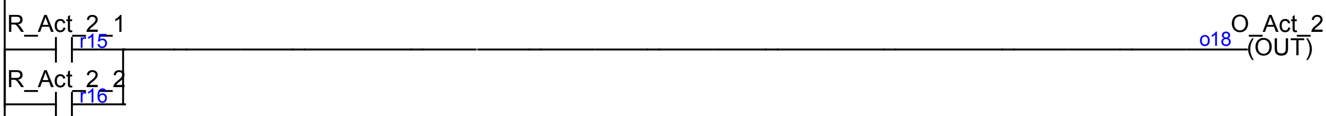
25 * First "on" time

26 

27 * Second "on" time (the off time occurs by logical magic)

28 

29 * Turn on physical output when necessary

30 

31 * Clear the Dispense2 relay once all the counters are finished, or clear the latch immediately if there's no drink in the slot
* The OR condition here will also fix the solenoid if it gets half a push out of time and the slot empties (drink drops on first push and then registers as empty)

32 

33

```
==========================================================================================
    _        _              _            _____
   / \   ___| |_ _   _  __ _| |_ ___  _ __ |___ /
  / _ \ / __| __| | | |/ _` | __/ _ \| '__|  |_ \
 / ___ \ (__| |_| |_| | (_| | || (_) | |    ___) |
/_/   \_\___|\__|\__,_|\__,_|\__\___/|_|   |____/

==========================================================================================
```

34 * First "on" time

35 R_Act_3 (r17) —| |— T_act_on_1 (t1) —|/|— T_act_off_1 (t2) —|/|— T_act_on_2 (t3) —|/|— (r18) R_Act_3_1 (RLY)

36 * Second "on" time (the off time occurs by logical magic)

37 R_Act_3 (r17) —| |— T_act_on_1 (t1) —| |— T_act_off_1 (t2) —| |— T_act_on_2 (t3) —|/|— (r19) R_Act_3_2 (RLY)

38 * Turn on physical output when necessary

39 R_Act_3_1 (r18) —| |— OR R_Act_3_2 (r19) —| |— (o19) O_Act_3 (OUT)

40 * Clear the Dispense3 relay once all the counters are finished,
  or clear the latch immediately if there's no drink in the slot
* The OR condition here will also fix the solenoid if it gets half a push
  out of time and the slot empties (drink drops on first
  push and then registers as empty)

41 R_Act_3 (r17) —| |— T_act_on_1 (t1) —| |— T_act_off_1 (t2) —| |— T_act_on_2 (t3) —| |— T_BCL3 (t6) —| |— (r17) R_Act_3 [Clear]; (r43) R_BCL3 [Clear]; (r3) Dispense3 [Clear]

42

```
==========================================================================================
    _        _              _            _  _
   / \   ___| |_ _   _  __ _| |_ ___  _ __ | || |
  / _ \ / __| __| | | |/ _` | __/ _ \| '__| | || |_
 / ___ \ (__| |_| |_| | (_| | || (_) | |    |__   _|
/_/   \_\___|\__|\__,_|\__,_|\__\___/|_|       |_|

==========================================================================================
```

43 * First "on" time

44 R_Act_4 (r20) —| |— T_act_on_1 (t1) —|/|— T_act_off_1 (t2) —|/|— T_act_on_2 (t3) —|/|— (r21) R_Act_4_1 (RLY)

45 * Second "on" time (the off time occurs by logical magic)

46 R_Act_4 (r20) —| |— T_act_on_1 (t1) —| |— T_act_off_1 (t2) —| |— T_act_on_2 (t3) —|/|— R_Act_4_2 (r22) (RLY)

47 * Turn on physical output when necessary

48 R_Act_4_1 (r21) —| |— OR R_Act_4_2 (r22) —| |— O_Act_4 (o20) (OUT)

49 * Clear the Dispense4 relay once all the counters are finished,
  or clear the latch immediately if there's no drink in the slot
  * The OR condition here will also fix the solenoid if it gets half a push
  out of time and the slot empties (drink drops on first
  push and then registers as empty)

50 R_Act_4 (r20) —| |— T_act_on_1 (t1) —| |— T_act_off_1 (t2) —| |— T_act_on_2 (t3) —| |— T_BCL4 (t7) —| |— R_Act_4 (r20) [Clear]; R_BCL4 (r44) [Clear]; Dispense4 (r4) [Clear]

51
```
================================================================================================
    _        _                _               ____  
   / \   ___| |_ _   _  __ _| |_ ___  _ __  | ___| 
  / _ \ / __| __| | | |/ _` | __/ _ \| '__| |___ \ 
 / ___ \ (__| |_| |_| | (_| | || (_) | |     ___) |
/_/   \_\___|\__|\__,_|\__,_|\__\___/|_|    |____/ 

================================================================================================
```

52 * First "on" time

53 R_Act_5 (r23) —| |— T_act_on_1 (t1) —|/|— T_act_off_1 (t2) —|/|— T_act_on_2 (t3) —|/|— R_Act_5_1 (r24) (RLY)

54 * Second "on" time (the off time occurs by logical magic)

55 R_Act_5 (r23) —| |— T_act_on_1 (t1) —| |— T_act_off_1 (t2) —| |— T_act_on_2 (t3) —|/|— R_Act_5_2 (r25) (RLY)

56 * Turn on physical output when necessary

57 R_Act_5_1 (r24) —| |— OR R_Act_5_2 (r25) —| |— O_Act_5 (o21) (OUT)

58 * Clear the Dispense5 relay once all the counters are finished,
  or clear the latch immediately if there's no drink in the slot
  * The OR condition here will also fix the solenoid if it gets half a push
  out of time and the slot empties (drink drops on first
  push and then registers as empty)

```
59  R_Act_5   T_act_on_1  T_act_off_1  T_act_on_2  T_BCL5                 R_Act_5
    --| |r23----| |t1-------| |t2---------| |t3-------| |t8-------------+--r23--[Clear]
                                                                       |       R_BCL5
                                                                       +--r45--[Clear]
                                                                       |       Dispense5
                                                                       +--r5---[Clear]
```

60
```
===========================================================================================
    _        _               _               __
   / \   ___| |_ _   _  __ _| |_ ___  _ __  / /_
  / _ \ / __| __| | | |/ _` | __/ _ \| '__|| '_ \
 / ___ \ (__| |_| |_| | (_| | || (_) | |   | (_) |
/_/   \_\___|\__|\__,_|\__,_|\__\___/|_|    \___/

===========================================================================================
```

61 * First "on" time

```
62  R_Act_6   T_act_on_1  T_act_off_1  T_act_on_2                          R_Act_6_1
    --| |r26----|/|t1-------|/|t2---------|/|t3----------------------------r27--(RLY)
```

63 * Second "on" time (the off time occurs by logical magic)

```
64  R_Act_6   T_act_on_1  T_act_off_1  T_act_on_2                          R_Act_6_2
    --| |r26----| |t1-------| |t2---------|/|t3----------------------------r28--(RLY)
```

65 * Turn on physical output when necessary

```
66  R_Act_6_1                                                              O_Act_6
    +--| |r27--+-----------------------------------------------------------o22--(OUT)
    |R_Act_6_2 |
    +--| |r28--+
```

67 * Clear the Dispense6 relay once all the counters are finished,
   or clear the latch immediately if there's no drink in the slot
 * The OR condition here will also fix the solenoid if it gets half a push
   out of time and the slot empties (drink drops on first
   push and then registers as empty)

```
68  R_Act_6   T_act_on_1  T_act_off_1  T_act_on_2  T_BCL6                 R_Act_6
    --| |r26----| |t1-------| |t2---------| |t3-------| |t9-------------+--r26--[Clear]
                                                                       |       R_BCL6
                                                                       +--r46--[Clear]
                                                                       |       Dispense6
                                                                       +--r6---[Clear]
```

69
```
===========================================================================================
    _        _               _               _____
   / \   ___| |_ _   _  __ _| |_ ___  _ __  |___  |
  / _ \ / __| __| | | |/ _` | __/ _ \| '__|    / /
 / ___ \ (__| |_| |_| | (_| | || (_) | |      / /
/_/   \_\___|\__|\__,_|\__,_|\__\___/|_|     /_/

===========================================================================================
```

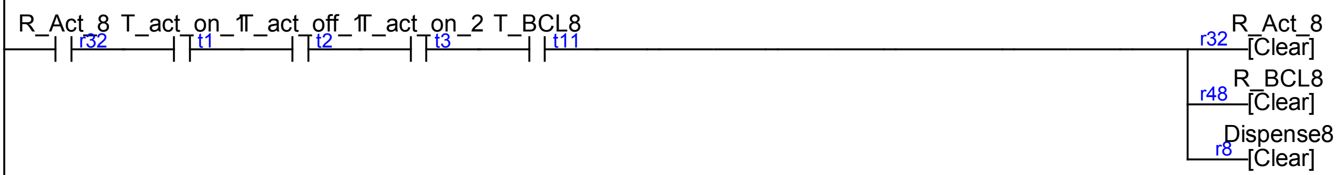
70 * First "on" time

```
71  R_Act_7    T_act_on_1   T_act_off_1   T_act_on_2                         R_Act_7_1
    --| |r29----|/|t1--------|/|t2---------|/|t3------------------------r30--(RLY)

72  * Second "on" time (the off time occurs by logical magic)

73  R_Act_7    T_act_on_1   T_act_off_1   T_act_on_2                         R_Act_7_2
    --| |r29----| |t1--------| |t2---------|/|t3------------------------r31--(RLY)

74  * Turn on physical output when necessary

75  R_Act_7_1                                                                O_Act_7
    --| |r30--+---------------------------------------------------------o23--(OUT)
    R_Act_7_2 |
    --| |r31--+

76  * Clear the Dispense7 relay once all the counters are finished,
      or clear the latch immediately if there's no drink in the slot
    * The OR condition here will also fix the solenoid if it gets half a
      push out of time and the slot empties (drink drops on first
      push and then registers as empty)

77  R_Act_7    T_act_on_1   T_act_off_1   T_act_on_2   T_BCL7                R_Act_7
    --| |r29----| |t1--------| |t2---------| |t3---------| |t10--------+-r29--[Clear]
                                                                      |      R_BCL7
                                                                      +-r47--[Clear]
                                                                      |      Dispense7
                                                                      +-r7---[Clear]

78  ==========================================================================================
        _            _              _                 ___
       / \    ___| |_ _   _  __ _| |_ ___  _ __   ( _ )
      / _ \  / __| __| | | |/ _` | __/ _ \| '__|  / _ \
     / ___ \ (__| |_| |_| | (_| | || (_) | |    | (_) |
    /_/   \_\___|\__|\__,_|\__,_|\__\___/|_|     \___/

    ==========================================================================================

79  * First "on" time

80  R_Act_8    T_act_on_1   T_act_off_1   T_act_on_2                         R_Act_8_1
    --| |r32----|/|t1--------|/|t2---------|/|t3------------------------r33--(RLY)

81  * Second "on" time (the off time occurs by logical magic)

82  R_Act_8    T_act_on_1   T_act_off_1   T_act_on_2                         R_Act_8_2
    --| |r32----| |t1--------| |t2---------|/|t3------------------------r34--(RLY)

83  * Turn on physical output when necessary

84  R_Act_8_1                                                                O_Act_8
    --| |r33--+---------------------------------------------------------o24--(OUT)
    R_Act_8_2 |
    --| |r34--+
```

85

* Clear the Dispense8 relay once all the counters are finished,
  or clear the latch immediately if there's no drink in the slot
* The OR condition here will also fix the solenoid if it gets half a
  push out of time and the slot empties (drink drops on first
  push and then registers as empty)

86



87

==

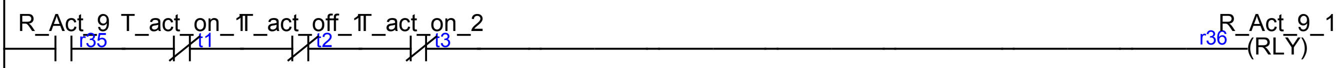
Actuator 9

==

88

* First "on" time

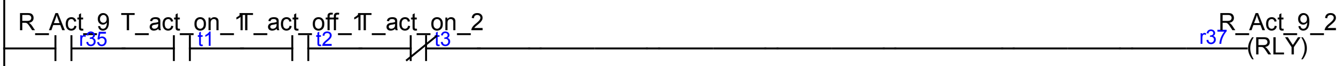
89



90

* Second "on" time (the off time occurs by logical magic)

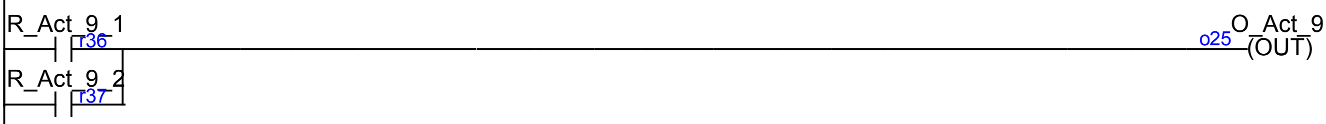
91



92

* Turn on physical output when necessary

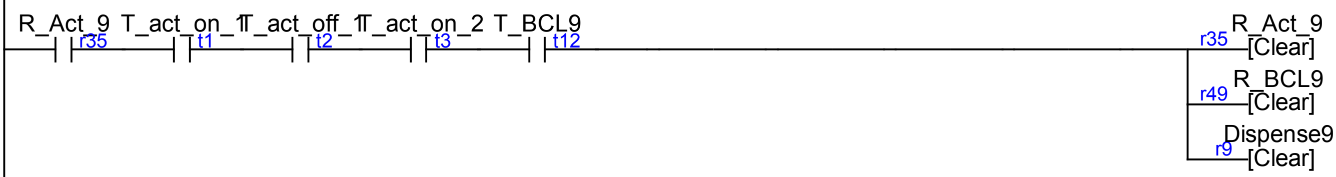
93



94

* Clear the Dispense9 relay once all the counters are finished,
  or clear the latch immediately if there's no drink in the slot
* The OR condition here will also fix the solenoid if it gets half a
  push out of time and the slot empties (drink drops on first
  push and then registers as empty)

95



96

```
==========================================================================================
    _        _                _               _  ___
   / \   ___| |_ _   _  __ _| |_ ___  _ __  / |/ _ \
  / _ \ / __| __| | | |/ _` | __/ _ \| '__| | | | | |
 / ___ \ (__| |_| |_| | (_| | || (_) | |    | | |_| |
/_/   \_\___|\__|\__,_|\__,_|\__\___/|_|    |_|\___/

==========================================================================================
```

97 * First "on" time

98
```
R_Act_10  T_act_on_1  T_act_off_1  T_act_on_2                                R_Act_10_1
--| |r38----|/|t1------|/|t2--------|/|t3-------------------------------r39--(RLY)
```

99 * Second "on" time (the off time occurs by logical magic)

100
```
R_Act_10  T_act_on_1  T_act_off_1  T_act_on_2                                R_Act_10_2
--| |r38----| |t1------| |t2--------|/|t3-------------------------------r40--(RLY)
```

101 * Turn on physical output when necessary

102
```
R_Act_10_1                                                                   O_Act_10
--+--| |r39--+------------------------------------------------------o26--(OUT)
  |R_Act_10_2|
  +--| |r40--+
```

103
* Clear the Dispense7 relay once all the counters are finished,
  or clear the latch immediately if there's no drink in the slot
* The OR condition here will also fix the solenoid if it gets half a
  push out of time and the slot empties (drink drops on first
  push and then registers as empty)

104
```
R_Act_10  T_act_on_1  T_act_off_1  T_act_on_2  T_BCL10                        R_Act_10
--| |r38----| |t1------| |t2--------| |t3------| |t13-----------------+--r38--[Clear]
                                                                     |        R_BCL10
                                                                     +--r50--[Clear]
                                                                     |        Dispense10
                                                                     +--r10--[Clear]
```

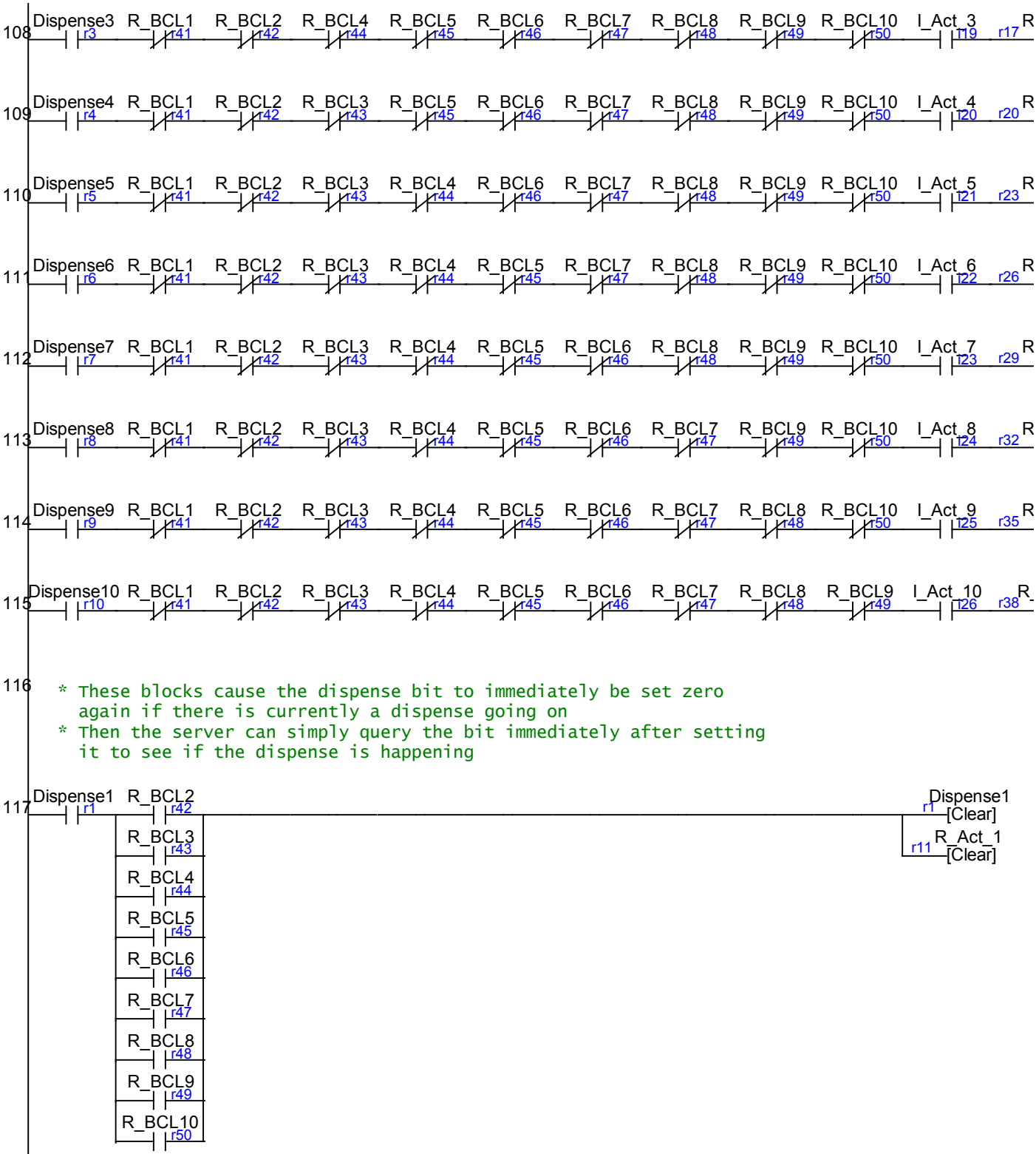
105

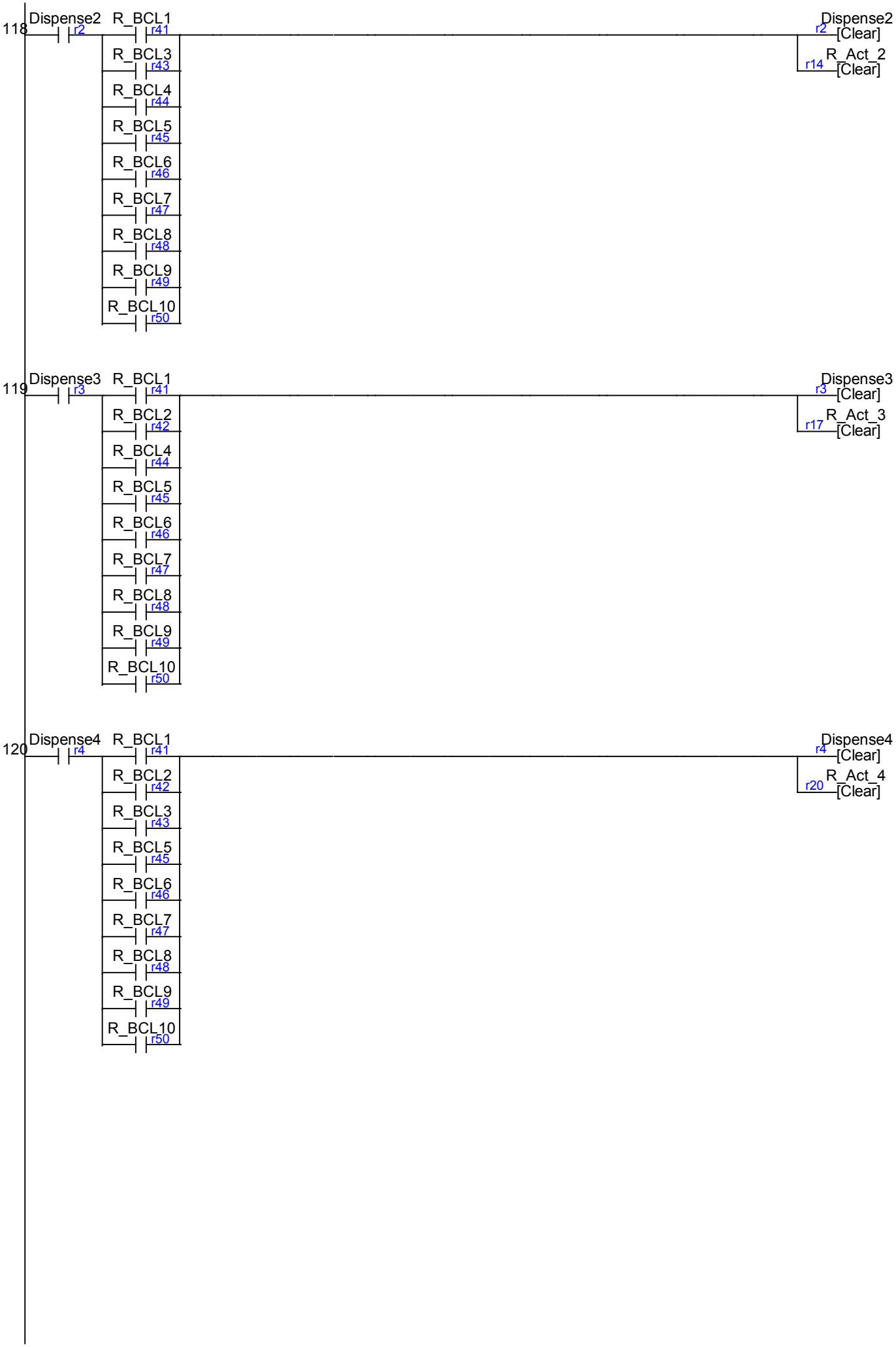
```
==========================================================================================
 _               _          ___      ____ _               _
| |    ___   ___| | _____  ( _ )    / ___| |__   ___  ___| | _____
| |   / _ \ / __| |/ / __| / _ \/\ | |   | '_ \ / _ \/ __| |/ / __|
| |__| (_) | (__|   <\__ \| (_>  < | |___| | | |  __/ (__|   <\__ \
|_____\___/ \___|_|\_\___/ \___/\/  \____|_| |_|\___|\___|_|\_\___/

==========================================================================================
```

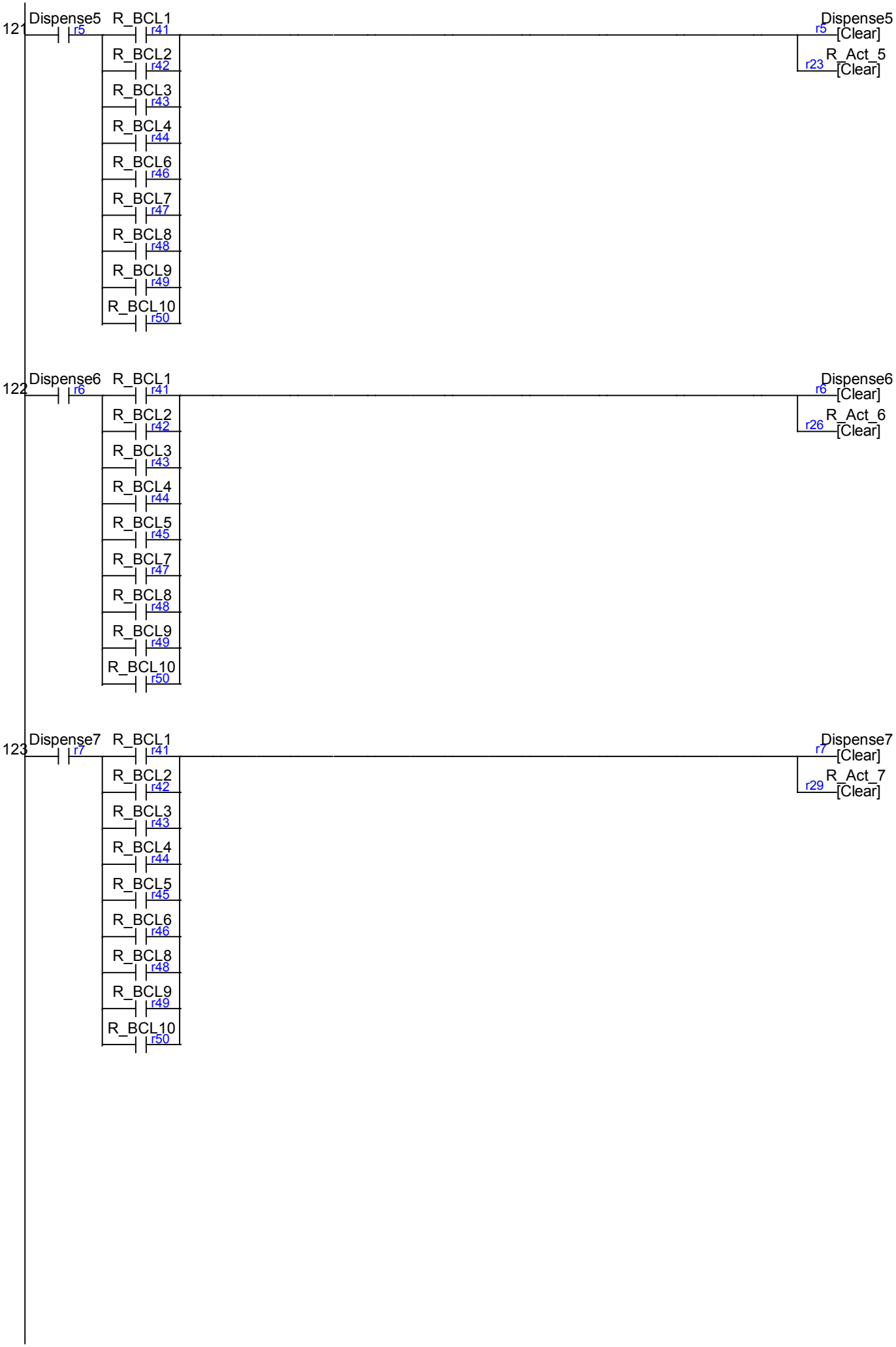
* Checking to make sure we can't dispense two things at the same time
* Only set the actuator latch high if there's nothing else dispensing
  and there's something in the slot at the start

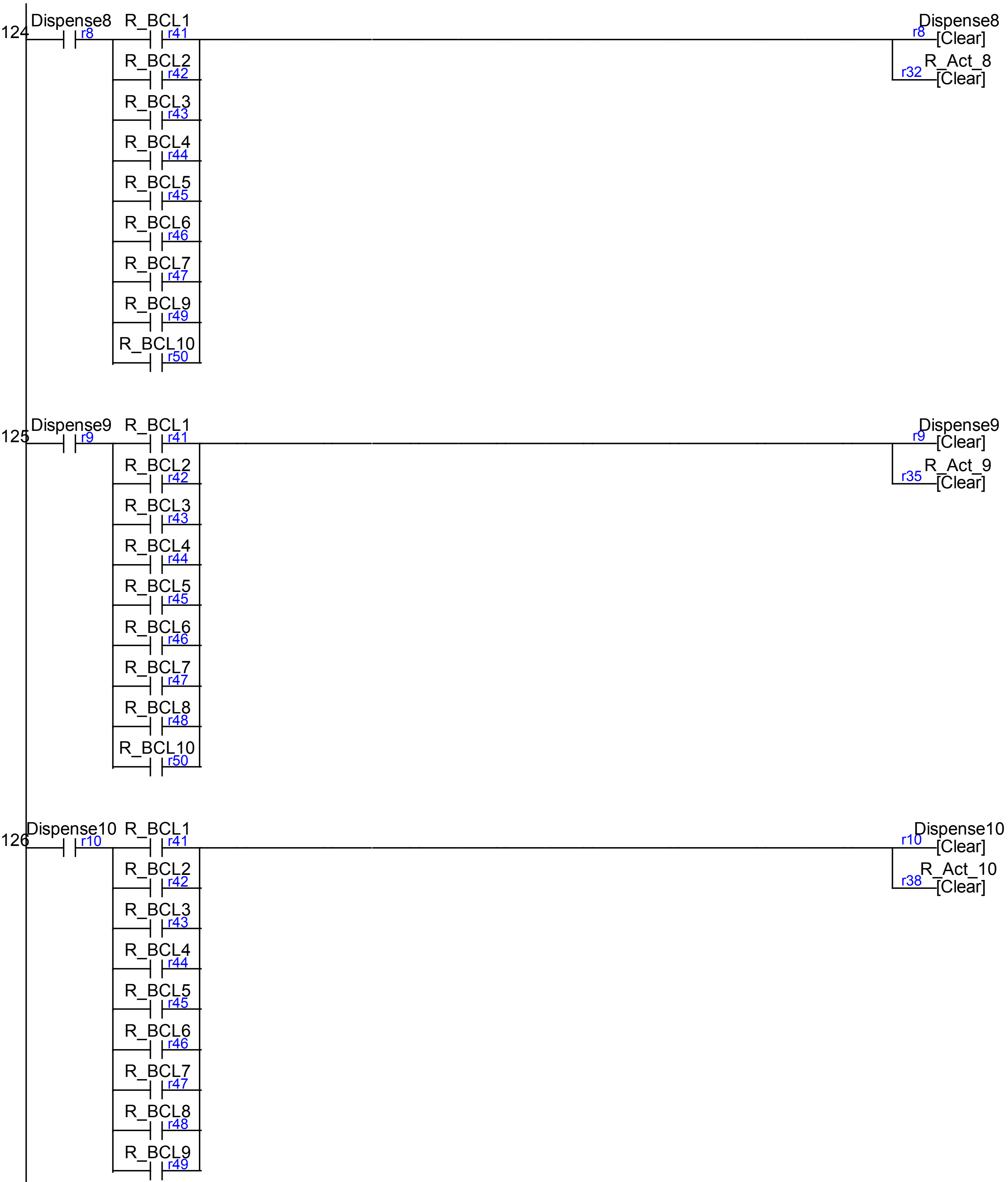
106
```
Dispense1  R_BCL2   R_BCL3   R_BCL4   R_BCL5   R_BCL6   R_BCL7   R_BCL8   R_BCL9   R_BCL10  I_Act_1       R
--| |r1----|/|r42---|/|r43---|/|r44---|/|r45---|/|r46---|/|r47---|/|r48---|/|r49---|/|r50---| |i17--r11--
```

107
```
Dispense2  R_BCL1   R_BCL3   R_BCL4   R_BCL5   R_BCL6   R_BCL7   R_BCL8   R_BCL9   R_BCL10  I_Act_2       R
--| |r2----|/|r41---|/|r43---|/|r44---|/|r45---|/|r46---|/|r47---|/|r48---|/|r49---|/|r50---| |i18--r14--
```


108 Dispense3 r3 R_BCL1 r41 R_BCL2 r42 R_BCL4 r44 R_BCL5 r45 R_BCL6 r46 R_BCL7 r47 R_BCL8 r48 R_BCL9 r49 R_BCL10 r50 I_Act_3 r19 R r17
109 Dispense4 r4 R_BCL1 r41 R_BCL2 r42 R_BCL3 r43 R_BCL5 r45 R_BCL6 r46 R_BCL7 r47 R_BCL8 r48 R_BCL9 r49 R_BCL10 r50 I_Act_4 r20 R r20
110 Dispense5 r5 R_BCL1 r41 R_BCL2 r42 R_BCL3 r43 R_BCL4 r44 R_BCL6 r46 R_BCL7 r47 R_BCL8 r48 R_BCL9 r49 R_BCL10 r50 I_Act_5 r21 R r23
111 Dispense6 r6 R_BCL1 r41 R_BCL2 r42 R_BCL3 r43 R_BCL4 r44 R_BCL5 r45 R_BCL7 r47 R_BCL8 r48 R_BCL9 r49 R_BCL10 r50 I_Act_6 r22 R r26
112 Dispense7 r7 R_BCL1 r41 R_BCL2 r42 R_BCL3 r43 R_BCL4 r44 R_BCL5 r45 R_BCL6 r46 R_BCL8 r48 R_BCL9 r49 R_BCL10 r50 I_Act_7 r23 R r29
113 Dispense8 r8 R_BCL1 r41 R_BCL2 r42 R_BCL3 r43 R_BCL4 r44 R_BCL5 r45 R_BCL6 r46 R_BCL7 r47 R_BCL9 r49 R_BCL10 r50 I_Act_8 r24 R r32
114 Dispense9 r9 R_BCL1 r41 R_BCL2 r42 R_BCL3 r43 R_BCL4 r44 R_BCL5 r45 R_BCL6 r46 R_BCL7 r47 R_BCL8 r48 R_BCL10 r50 I_Act_9 r25 R r35
115 Dispense10 r10 R_BCL1 r41 R_BCL2 r42 R_BCL3 r43 R_BCL4 r44 R_BCL5 r45 R_BCL6 r46 R_BCL7 r47 R_BCL8 r48 R_BCL9 r49 I_Act_10 r26 R r38
116 * These blocks cause the dispense bit to immediately be set zero again if there is currently a dispense going on
* Then the server can simply query the bit immediately after setting it to see if the dispense is happening
117 Dispense1 r1 R_BCL2 r42
R_BCL3 r43
R_BCL4 r44
R_BCL5 r45
R_BCL6 r46
R_BCL7 r47
R_BCL8 r48
R_BCL9 r49
R_BCL10 r50
Dispense1 r1 [Clear]
R_Act_1 r11 [Clear]


118
Dispense2
r2
R_BCL1
r41
R_BCL3
r43
R_BCL4
r44
R_BCL5
r45
R_BCL6
r46
R_BCL7
r47
R_BCL8
r48
R_BCL9
r49
R_BCL10
r50
Dispense2
r2
[Clear]
R_Act_2
r14
[Clear]
119
Dispense3
r3
R_BCL1
r41
R_BCL2
r42
R_BCL4
r44
R_BCL5
r45
R_BCL6
r46
R_BCL7
r47
R_BCL8
r48
R_BCL9
r49
R_BCL10
r50
Dispense3
r3
[Clear]
R_Act_3
r17
[Clear]
120
Dispense4
r4
R_BCL1
r41
R_BCL2
r42
R_BCL3
r43
R_BCL5
r45
R_BCL6
r46
R_BCL7
r47
R_BCL8
r48
R_BCL9
r49
R_BCL10
r50
Dispense4
r4
[Clear]
R_Act_4
r20
[Clear]


121
Dispense5
r5
R_BCL1
r41
R_BCL2
r42
R_BCL3
r43
R_BCL4
r44
R_BCL6
r46
R_BCL7
r47
R_BCL8
r48
R_BCL9
r49
R_BCL10
r50
Dispense5
r5
[Clear]
R_Act_5
r23
[Clear]
122
Dispense6
r6
R_BCL1
r41
R_BCL2
r42
R_BCL3
r43
R_BCL4
r44
R_BCL5
r45
R_BCL7
r47
R_BCL8
r48
R_BCL9
r49
R_BCL10
r50
Dispense6
r6
[Clear]
R_Act_6
r26
[Clear]
123
Dispense7
r7
R_BCL1
r41
R_BCL2
r42
R_BCL3
r43
R_BCL4
r44
R_BCL5
r45
R_BCL6
r46
R_BCL8
r48
R_BCL9
r49
R_BCL10
r50
Dispense7
r7
[Clear]
R_Act_7
r29
[Clear]



127

Manual Dispense

```
* If someone is holding down the manual dispense button and
  presses a front button, trigger a dispense
* Checking if there is something in the slot stops the edge case
  where the dispense bit could be manually set if there was nothing in the slot.
  This caused a can to dispense the next time one was put in the machine.
```

128 I_Button_1 (i1) —| |— I_Man_Disp (i30) —| |— I_Act_1 (i17) —| |— Dispense1 (r1) [Latch]

129 I_Button_2 (i2) —| |— I_Man_Disp (i30) —| |— I_Act_2 (i18) —| |— Dispense2 (r2) [Latch]

130 I_Button_3 (i3) —| |— I_Man_Disp (i30) —| |— I_Act_3 (i19) —| |— Dispense3 (r3) [Latch]

131 I_Button_4 (i4) —| |— I_Man_Disp (i30) —| |— I_Act_4 (i20) —| |— Dispense4 (r4) [Latch]

132 I_Button_5 (i5) —| |— I_Man_Disp (i30) —| |— I_Act_5 (i21) —| |— Dispense5 (r5) [Latch]

133 I_Button_6 (i6) —| |— I_Man_Disp (i30) —| |— I_Act_6 (i22) —| |— Dispense6 (r6) [Latch]

134 I_Button_7 (i7) —| |— I_Man_Disp (i30) —| |— I_Act_7 (i23) —| |— Dispense7 (r7) [Latch]

135 * The last three buttons are different to the other buttons in
the machine - normally closed

136 I_Button_8 (i27) —|/|— I_Man_Disp (i30) —| |— I_Act_8 (i24) —| |— Dispense8 (r8) [Latch]

137 I_Button_9 (i28) —|/|— I_Man_Disp (i30) —| |— I_Act_9 (i25) —| |— Dispense9 (r9) [Latch]

138 I_Button_10 (i29) —|/|— I_Man_Disp (i30) —| |— I_Act_10 (i26) —| |— Dispense10 (r10) [Latch]

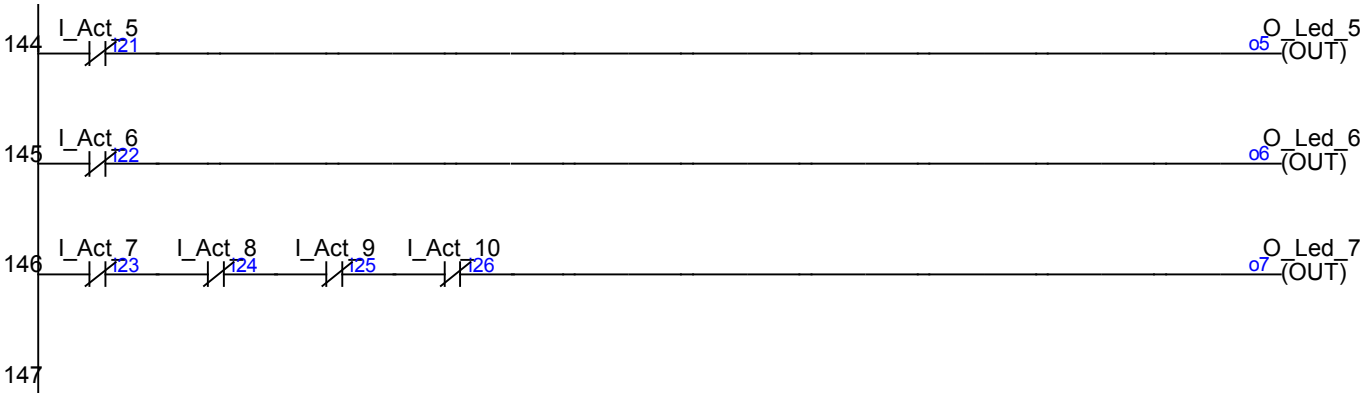
139
```
==========================================================================================
 _____                 _           _     _       _     _
| ____|_ __ ___  _ __ | |_ _   _  | |   (_) __ _| |__ | |_ ___
|  _| | '_ ` _ \| '_ \| __| | | | | |   | |/ _` | '_ \| __/ __|
| |___| | | | | | |_) | |_| |_| | | |___| | (_| | | | | |_\__ \
|_____|_| |_| |_| .__/ \__|\__, | |_____|_|\__, |_| |_|\__|___/
                |_|        |___/           |___/
==========================================================================================
```
* Turn on the lights at the front of the machine when the slots are empty
* Only turn on the led for button7 when all of actuators 7,8,9,10 are empty
* Normally closed hardware logic is used so that the slots will register as
empty if the switches ever fail

140 I_Act_1 (i17) —|/|— O_Led_1 (o1) (OUT)

141 I_Act_2 (i18) —|/|— O_Led_2 (o2) (OUT)

142 I_Act_3 (i19) —|/|— O_Led_3 (o3) (OUT)

143 I_Act_4 (i20) —|/|— O_Led_4 (o4) (OUT)


144
I_Act_5
i21
O_Led_5
o5
(OUT)
145
I_Act_6
i22
O_Led_6
o6
(OUT)
146
I_Act_7
i23
I_Act_8
i24
I_Act_9
i25
I_Act_10
i26
O_Led_7
o7
(OUT)
147