Cf. https://www.plctalk.net/qanda/showthread.php?t=130748

Minimal two-way traffic lights

Six sequential timers:
- 25s Green 2
- 5s Yellow 2
- 3s Overlap 2; Red 2 start and Red 1 end i.e. overlapped Reds
- 25s Green 1
- 5s Yellow 2
- 3s Overlap 1; Red 1 start and Red 2 end i.e. overlapped Reds

0000

OVERLAP_1 Done and Cycle Done
T4:5
DN

Nighttime
B3:0
0

GREEN_2
TON
Timer On Delay
Timer T4:0
Time Base 1.0
Preset 25<
Accum 0<
EN
DN

0001

GREEN_2 Done
T4:0
DN

YELLOW_2
TON
Timer On Delay
Timer T4:1
Time Base 1.0
Preset 5<
Accum 0<
EN
DN

0002

YELLOW_2 Done
T4:1
DN

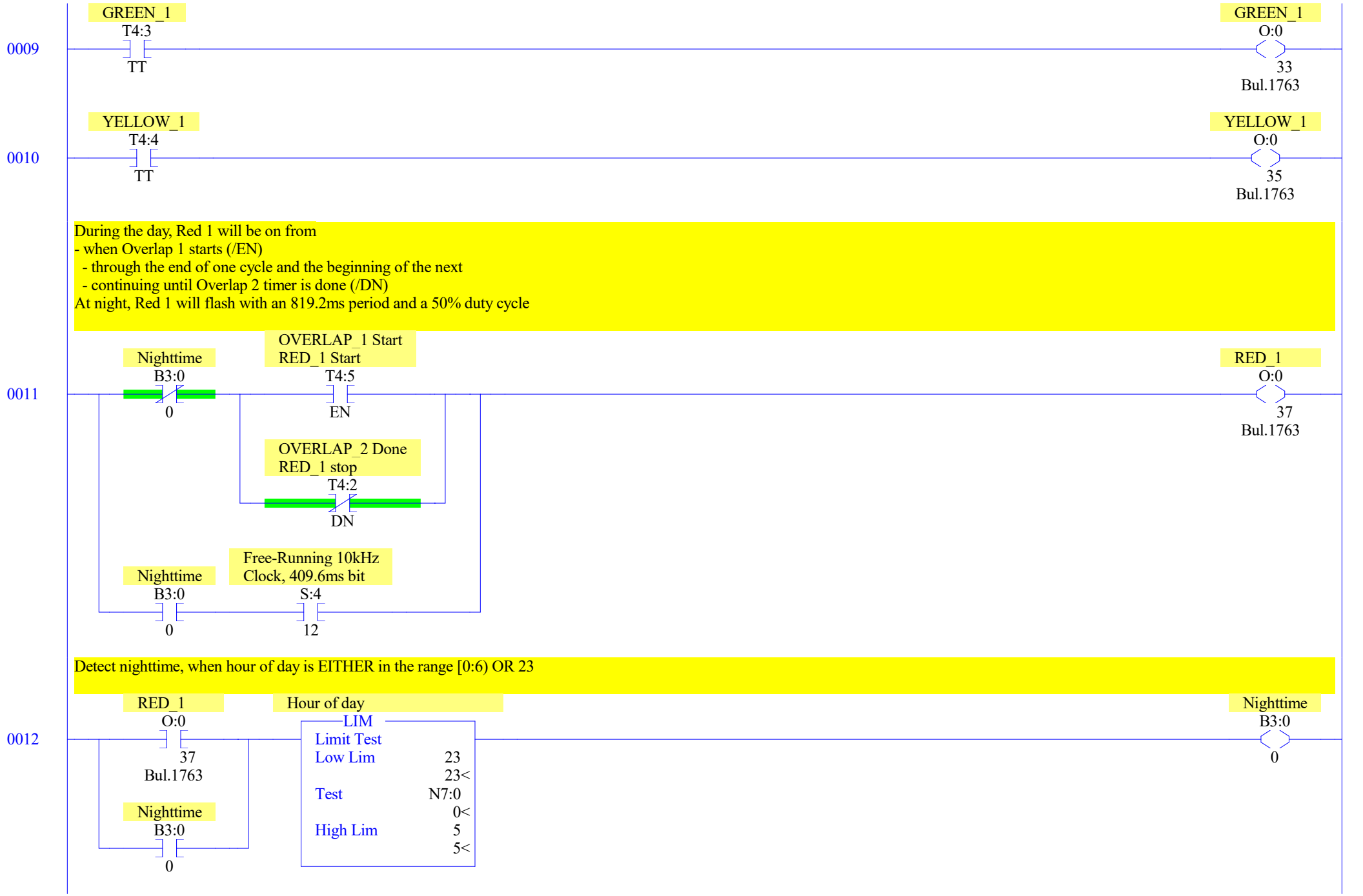
OVERLAP_2
TON
Timer On Delay
Timer T4:2
Time Base 1.0
Preset 3<
Accum 0<
EN
DN

0003

OVERLAP_2 Done
RED_1 stop
T4:2
DN

GREEN_1
TON
Timer On Delay
Timer T4:3
Time Base 1.0
Preset 25<
Accum 0<
EN
DN

GREEN_1 Done
T4:3
DN
0004
YELLOW_1
TON
Timer On Delay
Timer T4:4
Time Base 1.0
Preset 5<
Accum 0<
EN
DN
YELLOW_1 Done
T4:4
DN
0005
OVERLAP_1
TON
Timer On Delay
Timer T4:5
Time Base 1.0
Preset 3<
Accum 0<
EN
DN
During the day:
- All Green and Yellow lights will be on when their corresponding timers are timing (/TT)
See comments below for special cases at night
GREEN_2
T4:0
TT
0006
GREEN_2
O:0
42
Bul.1763
At night, Yellow 2 will flash with an 819.2ms period and a 50% duty cycle
YELLOW_2
T4:1
TT
0007
YELLOW_2
O:0
44
Bul.1763
Nighttime
B3:0
0
Free-Running 10kHz
Clock, 409.6ms bit
S:4
12
During the day, Red 2 will be on from when Overlap 2 starts (/EN) until the end of the six-timer cycle
RED_2 Start
T4:2
EN
0008
RED_2
O:0
46
Bul.1763


GREEN_1
T4:3
TT
GREEN_1
O:0
33
Bul.1763
0009
YELLOW_1
T4:4
TT
YELLOW_1
O:0
35
Bul.1763
0010
During the day, Red 1 will be on from
- when Overlap 1 starts (/EN)
- through the end of one cycle and the beginning of the next
- continuing until Overlap 2 timer is done (/DN)
At night, Red 1 will flash with an 819.2ms period and a 50% duty cycle
Nighttime
B3:0
0
OVERLAP_1 Start
RED_1 Start
T4:5
EN
RED_1
O:0
37
Bul.1763
0011
OVERLAP_2 Done
RED_1 stop
T4:2
DN
Free-Running 10kHz
Clock, 409.6ms bit
Nighttime
B3:0
0
S:4
12
Detect nighttime, when hour of day is EITHER in the range [0:6) OR 23
RED_1
O:0
37
Bul.1763
Hour of day
LIM
Limit Test
Low Lim
23
23<
Test
N7:0
0<
High Lim
5
5<
Nighttime
B3:0
0
Nighttime
B3:0
0
0012

Code to determine hour of day
- If N7:1 is outside the range [0:23], then use the hour value from the PLC Real-Time Clock (RTC:0.HR)
  - Default value of N7:1 is -1, so the RTC is the default source
- For testing, if N7:1 in the range [0:24), then use it as the hour of day
  - Also, at start of every odd minute-of-hour, increment N7:1, including wrap-around from 23 to 0

0013

Hour of day override if in range [0:23]

LIM
Limit Test
Low Lim 24 24<
Test N7:1 -1<
High Lim -1 -1<

Hour of day

MOV
Move
Source RTC:0.HR 0<
Dest N7:0 0<

Hour of day override if in range [0:23]

LIM
Limit Test
Low Lim 0 0<
Test N7:1 -1<
High Lim 23 23<

EQU
Equal
Source A RTC:0.SEC 0<
Source B 0 0<

Two-minute cycle
RTC:0.MIN
0

One-shot memory
B3:0
ONS
15

Hour of day override if in range [0:23]

ADD
Add
Source A N7:1 -1<
Source B 1 1<
Dest N7:1 -1<

Hour of day override if in range [0:23]

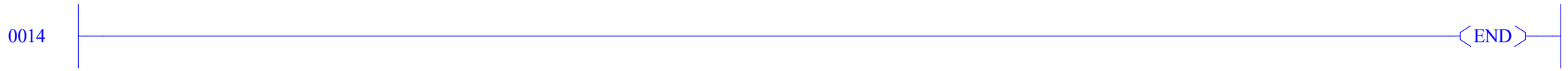
GRT
Greater Than (A>B)
Source A N7:1 -1<
Source B 23 23<

Hour of day override if in range [0:23]

MOV
Move
Source 0 0<
Dest N7:1 -1<

Hour of day

MOV
Move
Source N7:1 -1<
Dest N7:0 0<


0014
END