DTU Electrical Engineering

ASSIGNEMENT 4

31342 - Introduction to Programmable Logic Controllers

João Ramiro
S182059@student.dtu.dk

Part 1 – Logic Functions

Using the provided truth table the following equations were built:

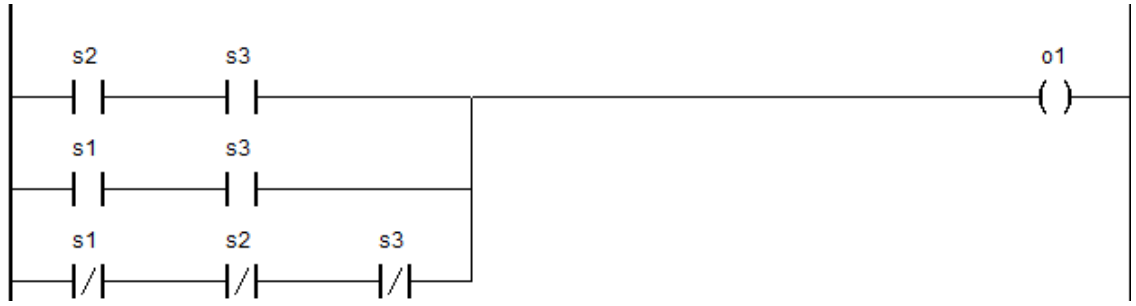
$$\text{Red} = S_2S_3 + S_1S_3 + \overline{S_1}\,\overline{S_3}\,\overline{S_3}$$

$$\text{Yellow} = (\overline{S_1} + \overline{S_3}) \cdot (S_1 + S_2 + S_3)$$

$$\text{Green} = (S_1 + \overline{S_2}) \cdot (\overline{S_2} + \overline{S_3})(\overline{S_1} + S_2 + S_3)$$

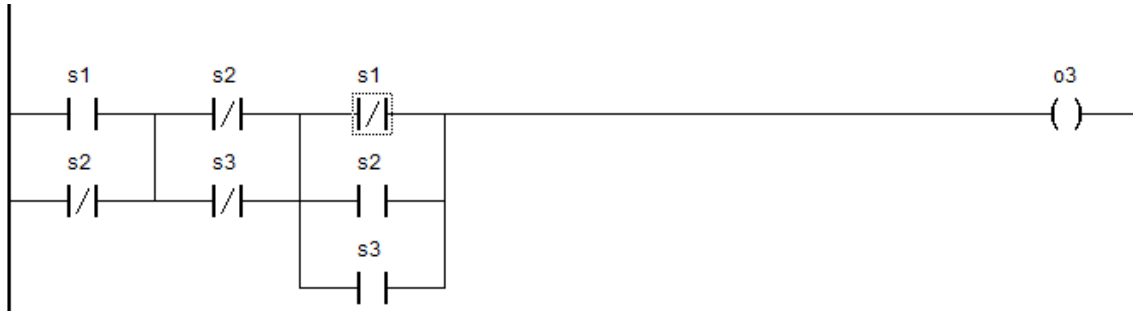
After that, the equations were used to build this ladder diagram:

Red:



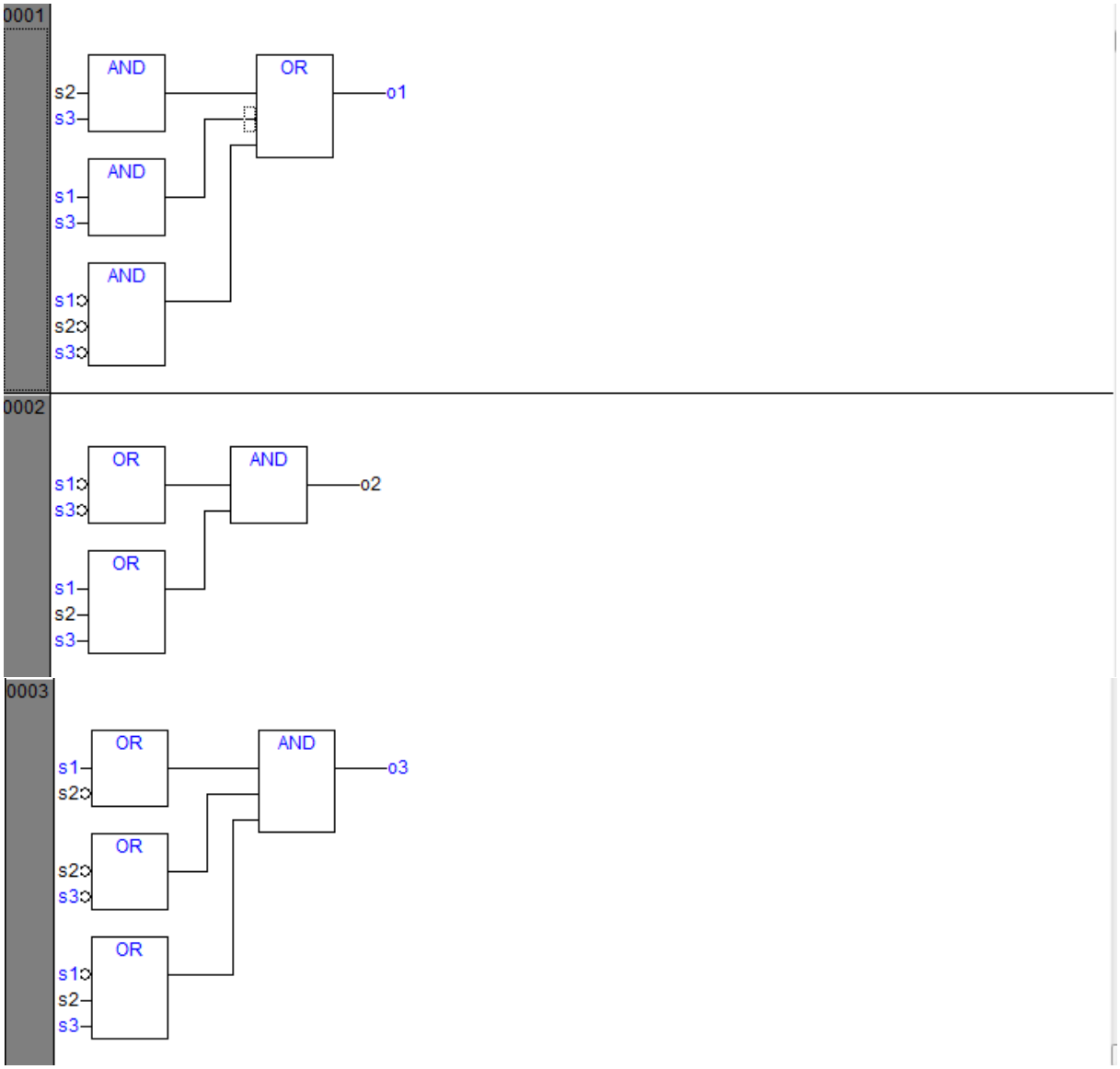
Yellow:

Green:



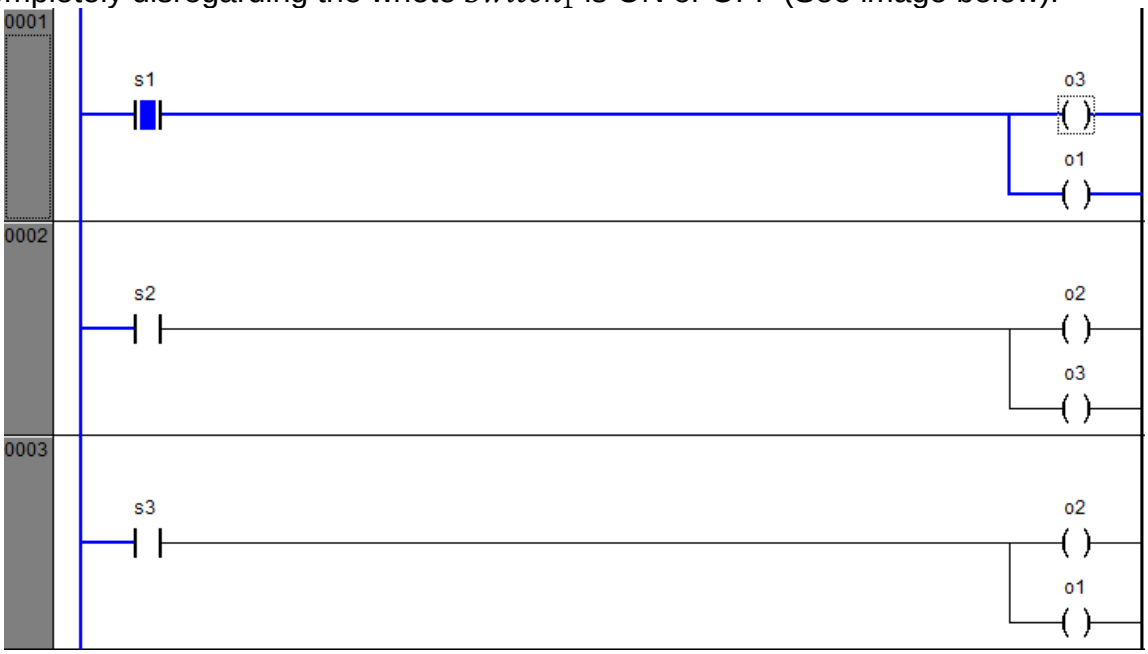
It can be found in attachment a video showing the implementation.

Finally, the ladder diagram was converted automatically into FDB, which generated the correct circuit seen below:

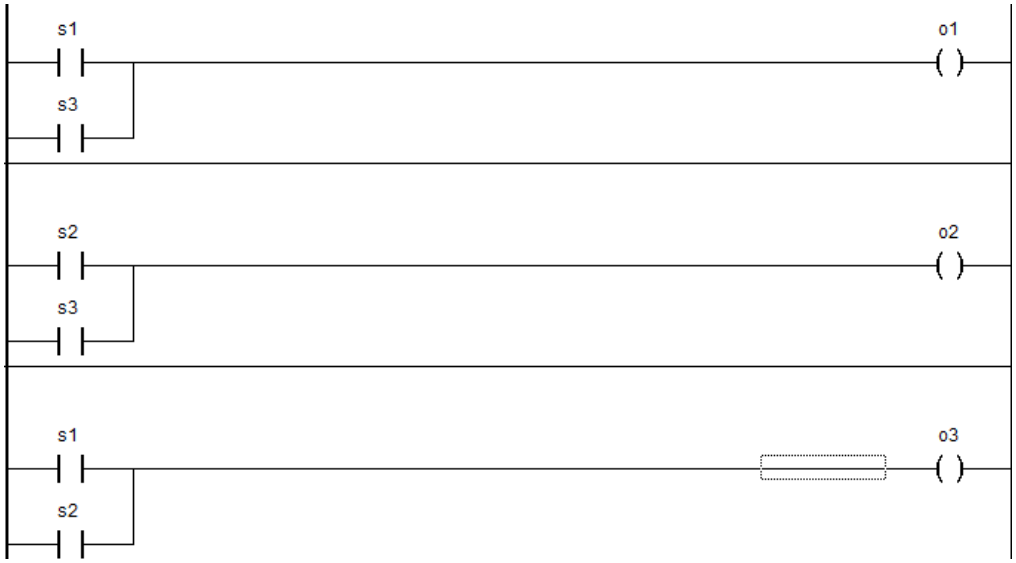


Part 2 - Priority of networks

By implementing the provided Ladder Diagram, we can see that the nets are executed in order, meaning nets that are executed later override previous ones. This means for example that if $Switch_1$ then $green$ and red should be ON, but since the in the next networks $Switch_2$ and $Switch_3$ are OFF they will set $green$ and red are OFF, thus completely disregarding the whete $Switch_1$ is ON or OFF (See image below).



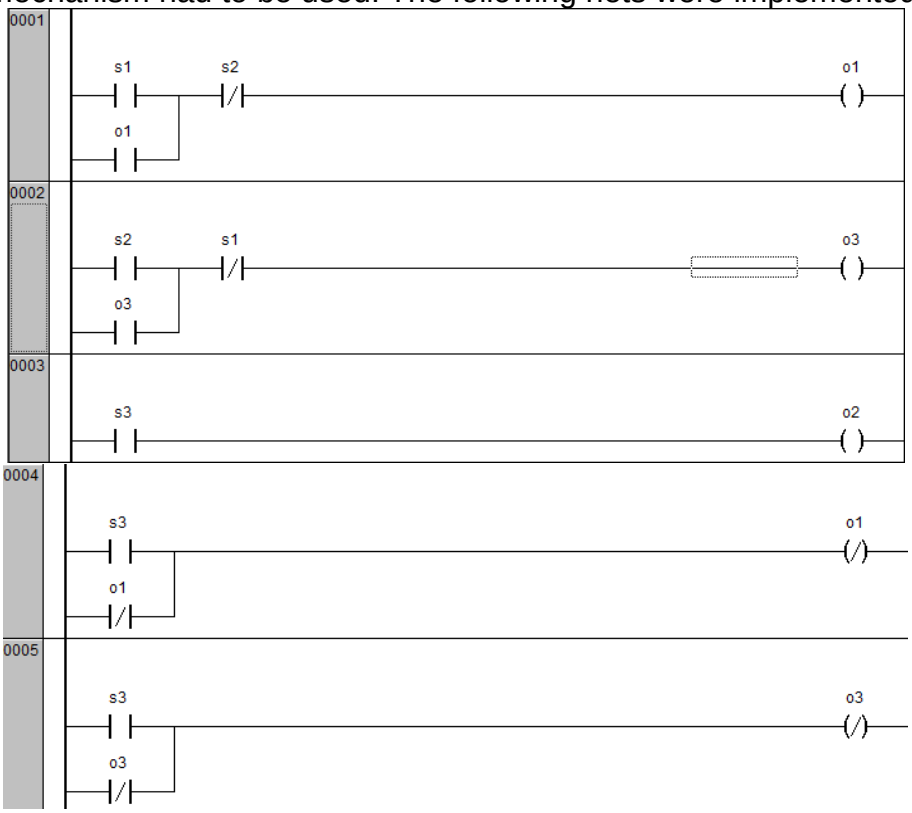
A better approach is, instead of each net having 1 input and multiple outputs, having multiple inputs and only 1 input. This will make it so that this kind of conflicts don't occur. In the image below it was chosen to implement that architecture using the OR:



Part 3 - Retentive Coils

System with normal coils only

To implement the system described in the assignement without normal coils the latch mechanism had to be used. The following nets were implemented:

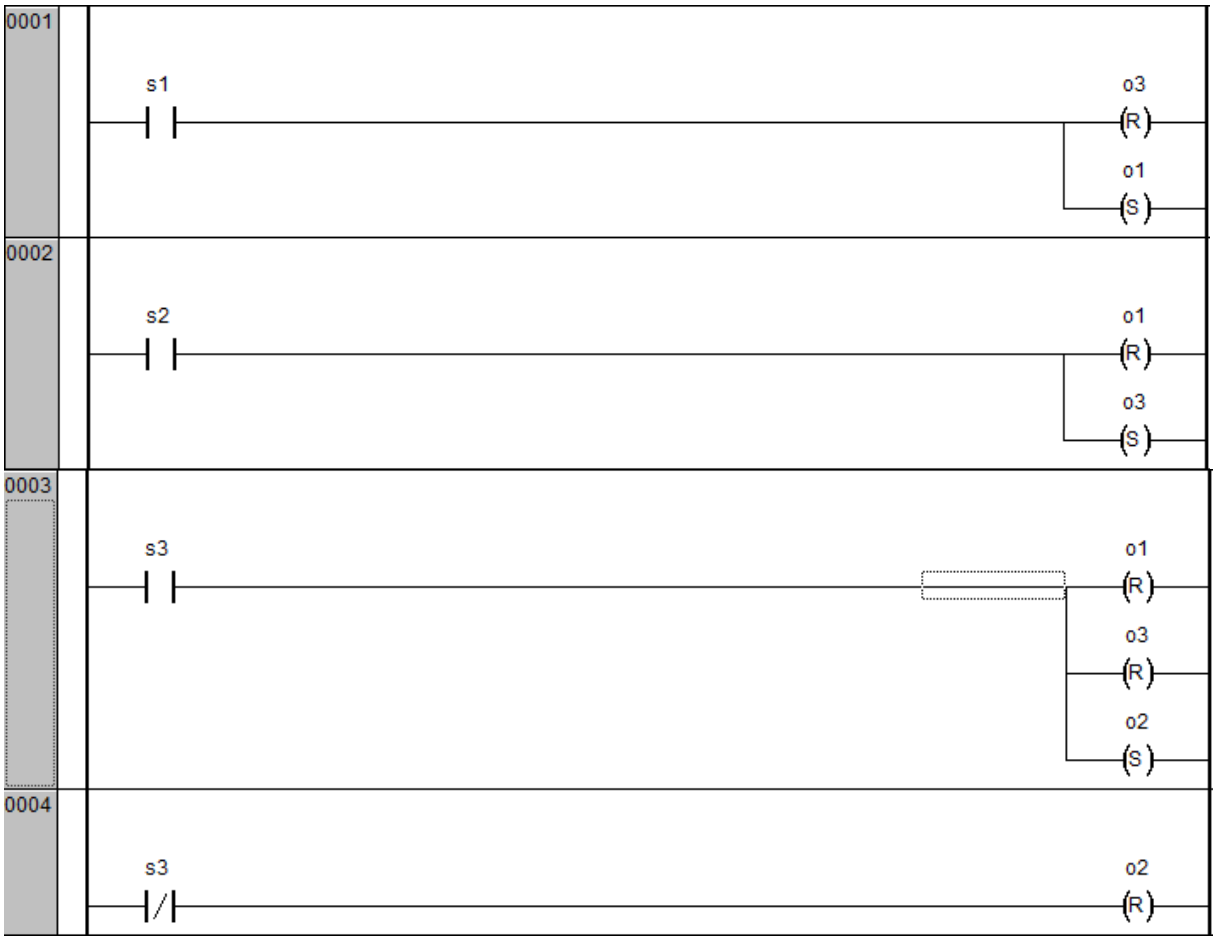


Description

- Net1 is used to turn on the Red light making the conveyor belt go right for this to happen switch1 must be triggered. To prevent any weird behaviour it is a requirement that switch2 must be OFF
- Net2 is used to turn on the Green light making the conveyor belt go left. To prevent any weird behaviour it is a requirement that switch1 must be OFF
- Net3 Net4 and Net5 are used to implement the emergency stop:
    - Net3 simply turns on the emergency light ON when the emergency button is pressed;
    - Net4 stops the conveyor belt from going right in case the emergency button is pressed.
    - Net5 stops the conveyor belt from going left in case the emergency button is pressed

System only with retentive coils

The schematic below shows how this system was implemented:



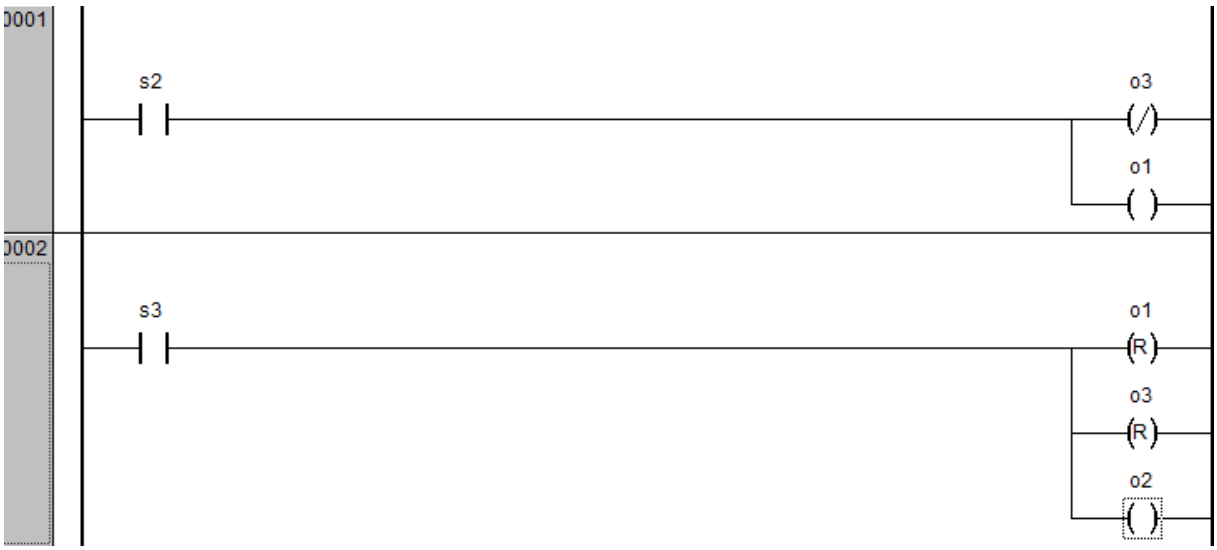
Description

- Net1 is used to make the conveyor belt go right.
- Net2 is used to make the conveyor belt go left.
- Net3 is used has the emergency button which will stop the conveyor belt since it overrides previous nets
- Net4 is used to turn the emergency button OFF.

Using retentive coils seems the best way to implement it since the coils keep the state without the need for latches.

System only one input switch (apart from the emergency stop)

Yes its possible to implement only using one input switch. It is actually easier to do so, since before the two switches had to, in a way, complement each other.
Using switch2 as a toggle switch and switch3 as the emergency stop button the following schematic was implemented:



Description

- Net1 is used to make the conveyor belt go left or right depending on the state of the switch.
- Net2 is used to stop the conveyor belt when the emergency button is triggered.