Exercise 2: RGB LED

1 Objective

This exercise has three parts.

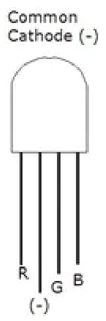
1. Build a circuit with the push button and RGB LED on a breadboard.
2. Write ladder logic to control the circuit
   a. Step through an 8-color sequence with two control modes
   b. mode 1: push button changes color
   c. mode 2: colors change cyclically with a 1s timer
3. Connect to an HMI.
   a. The HMI is provided. You will need to import it into ScadaBR and start it.

2 Install ScadaBR

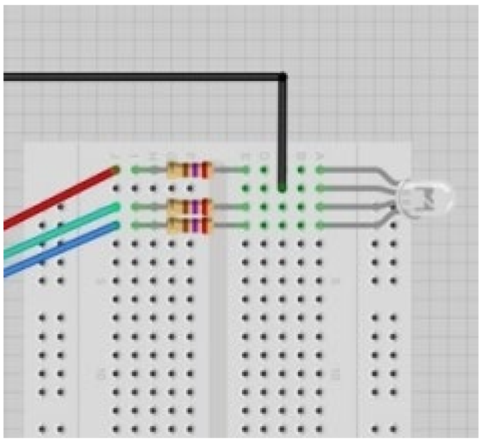
1. Use the instructions from the lecture.
   a. Install this Java IDK: https://adoptium.net/temurin/releases/?version=8
   b. Install ScadaBR: https://github.com/ScadaBR/ScadaBR/releases/tag/v1.2

3 Build the circuit

1. Insert the 3 color LED into the breadboard. The RGB LED has a common cathode pin. It is the longest pin on the device. Connect this pin to ground on the Arduino.



2. Connect the R, G, and B pins to a 1K Ohm resistor and connect the other end of the resistor to pins 7, 8, and 12 respectively.



3. Connect a button to pin 2 of the Arduino. Refer to exercise one for the appropriate button circuit.
4. Connect power and ground rails of the breadboard to the 5V and GND pins of the Arduino.

4 Write Ladder Logic

1. Using the OpenPLC Editor, create a new project.
2. Create the following variables.

Name	Class	Type	Location
count	Local	INT	%MW8
color0	Local	INT	%MW0
color1	Local	INT	%MW1
color2	Local	INT	%MW2
color3	Local	INT	%MW3
color4	Local	INT	%MW4
color5	Local	INT	%MW5
color6	Local	INT	%MW6
color7	Local	INT	%MW7
red_anode	Local	BOOL	%QX100.0
green_anode	Local	BOOL	%QX100.1
blue_anode	Local	BOOL	%QX100.2
button	Local	BOOL	%IX100.0
mode	Local	BOOL	%QX0.0

The 8 color variables (*color0-color7*) are connected to registers. These registers can be set by the HMI (ScadaBR) via MODBUS. This will allow the user to control what color is output for each count value. **Important:** color0-color7 bits must be interpreted on the Ladder Logic program as direct controls to red_anode (bit 0), green_anode (bit 1) and blue_anode (bit 2). For example, if color0 = 3, it means that red_anode and green_anode must be activated, and blue_anode is turned off.

count is also a register, but this variable will be controlled by ladder logic. The %MW8 is a read/write type and in this case is used to allow the HMI to read the count and display it. An attacker might choose to write to this variable!

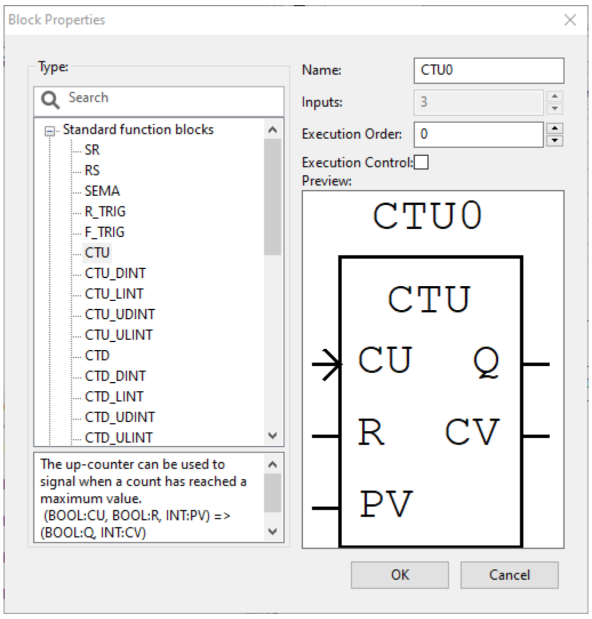
red_anode, *green_anode*, and *blue_anode* are controlled by the ladder logic. These values will drive the Arduino pins connected to the RGB anodes on the RGB LED.

button is an input from the Arduino. When the button is pressed this value will be 1. When the button is released this value will be 0.

mode is an input from the ScadaBR HMI. %QX0.0 is a single bit read/write coil. *mode* is not connected to the breadboard. Instead we will control it from the HMI.

Important: These are the mandatory variables for HMI-PLC sync. You certainly will need additional internal variables to accomplish your logic.

3. Design the Ladder Logic circuit. Since this is a more complex program, it is advisable to design it in steps. Test each step individually on the OpenPLC Simulator to make sure it is working as intended. The following steps guide might be helpful for you:
    a. Design a 1 second oscillator. This oscillator must produce a square wave with 500ms ON pulse and 500ms OFF pulse. This will be used to step through the 8-color sequence automatically when mode is set to zero.
    b. Use the CTU block to design a circuit that goes from 0 to 8. The circuit must auto reset when the count reaches 8


Block Properties

Type:

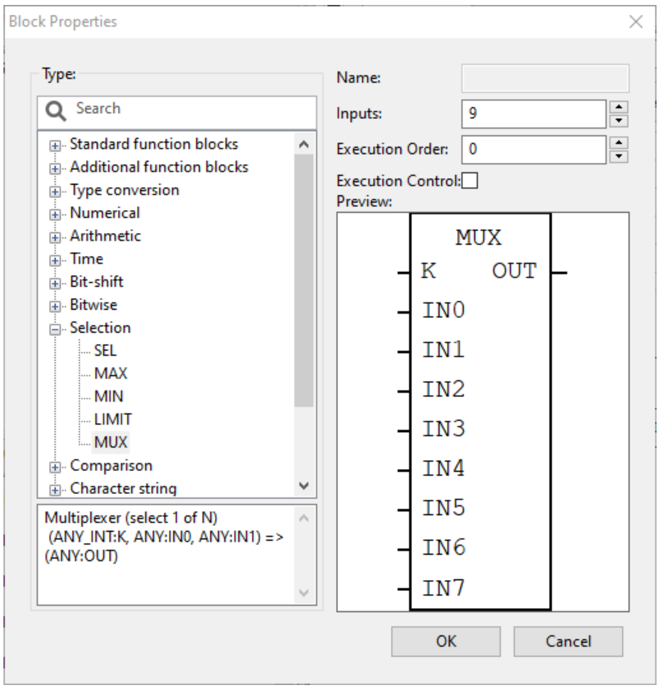
Search

Standard function blocks
- SR
- RS
- SEMA
- R_TRIG
- F_TRIG
- CTU
- CTU_DINT
- CTU_LINT
- CTU_UDINT
- CTU_ULINT
- CTD
- CTD_DINT
- CTD_LINT
- CTD_UDINT
- CTD_ULINT

The up-counter can be used to signal when a count has reached a maximum value.
(BOOL:CU, BOOL:R, INT:PV) => (BOOL:Q, INT:CV)

Name: CTU0
Inputs: 3
Execution Order: 0
Execution Control: ☐
Preview:

    c. Use the MUX block to design a circuit that selects one of the 8 possible inputs as the output. You will use this circuit to select one of the color0-color7 colors to display on the RGB LED.



Name:

Inputs: 9

Execution Order: 0

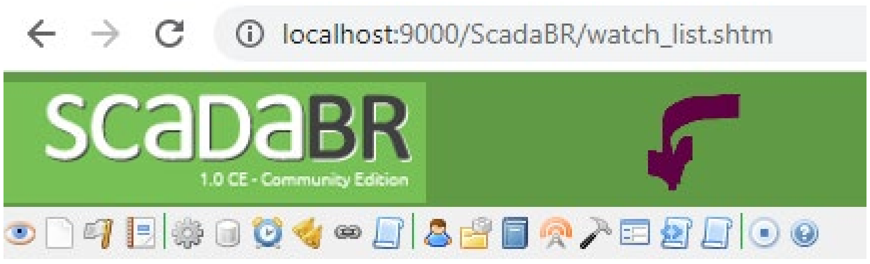
Execution Control: ☐

Multiplexer (select 1 of N)
(ANY_INT:K, ANY:IN0, ANY:IN1) => (ANY:OUT)

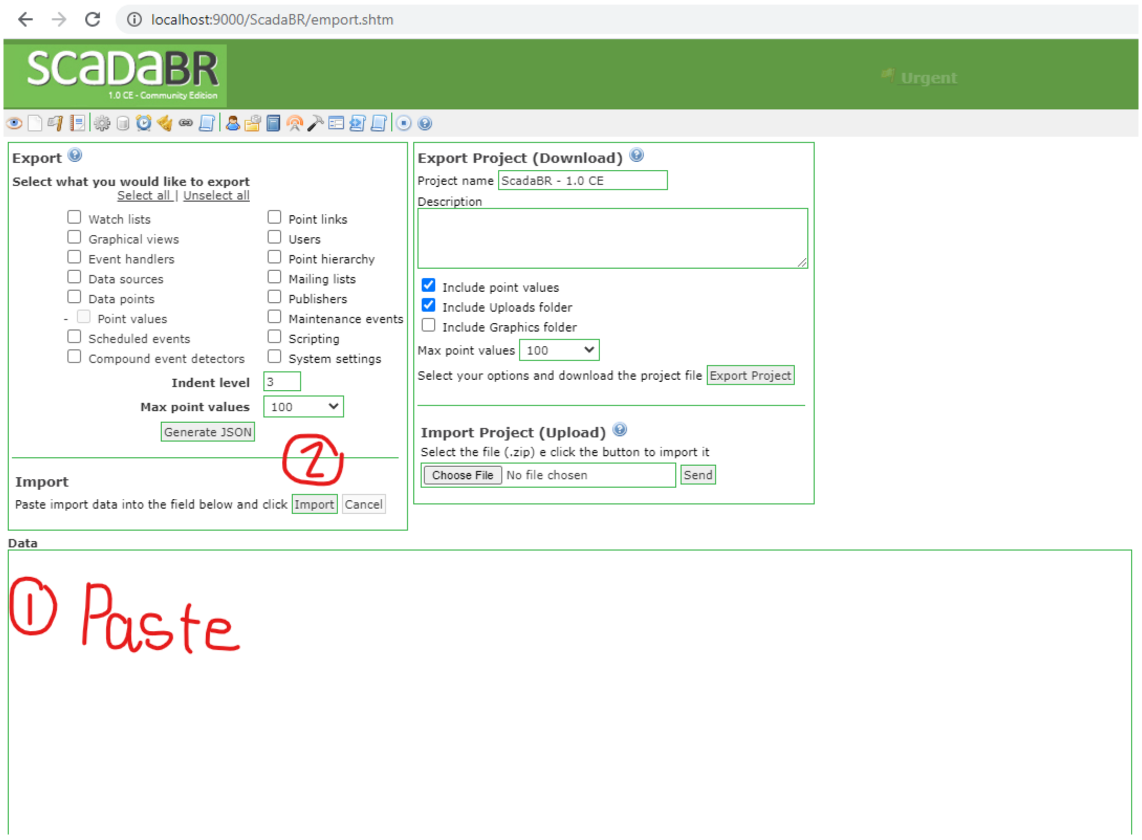
d. Design a circuit to extract the least significant 3 bits of a WORD. You can do this directly in your main Ladder Logic program, or design it as a Function Block that you can later reuse in your program. The main advantage of designing it as a Function Block is that you can pick a different language to design this logic. For this particular problem, Structured Text is way easier to implement than Ladder Logic. You will use this to select the least significant 3 bits of the MUX output from above and connect them to red_anode (bit 0), green_anode (bit 1), and blue_anode (bit 2).

5 Connect the HMI

1. We will not develop a new HMI for this exercise. Instead we will import an existing HMI.
2. The Exercise 2 folder on Piazza includes an exercise2-hmi.json file. Download this file to your PC.
3. Navigate to ScadaBR.
   a. ScadaBR should start automatically after booting the machine you installed it on.
   b. Navigate to ScadaBR in your host’s browser. http://[HMI VM IP ADDRESS]:8080/ScadaBR
   c. Login to ScadaBR (user: admin, password: admin)
4. Import the exercise2-hmi.json file.
   a. Click Import/Export on the toolbar.


localhost:9000/ScadaBR/watch_list.shtm
scadaBR
1.0 CE - Community Edition

b. Manually paste contents of the exercise2-hmi.json file into the data field.


localhost:9000/ScadaBR/emport.shtm
scadaBR
1.0 CE - Community Edition
Urgent
②
① Paste

Export
Select what you would like to export
Select all | Unselect all

☐ Watch lists	☐ Point links
☐ Graphical views	☐ Users
☐ Event handlers	☐ Point hierarchy
☐ Data sources	☐ Mailing lists
☐ Data points	☐ Publishers
- ☐ Point values	☐ Maintenance events
☐ Scheduled events	☐ Scripting
☐ Compound event detectors	☐ System settings

Indent level: 3
Max point values: 100
Generate JSON

Import
Paste import data into the field below and click Import Cancel

Export Project (Download)
Project name: ScadaBR - 1.0 CE
Description

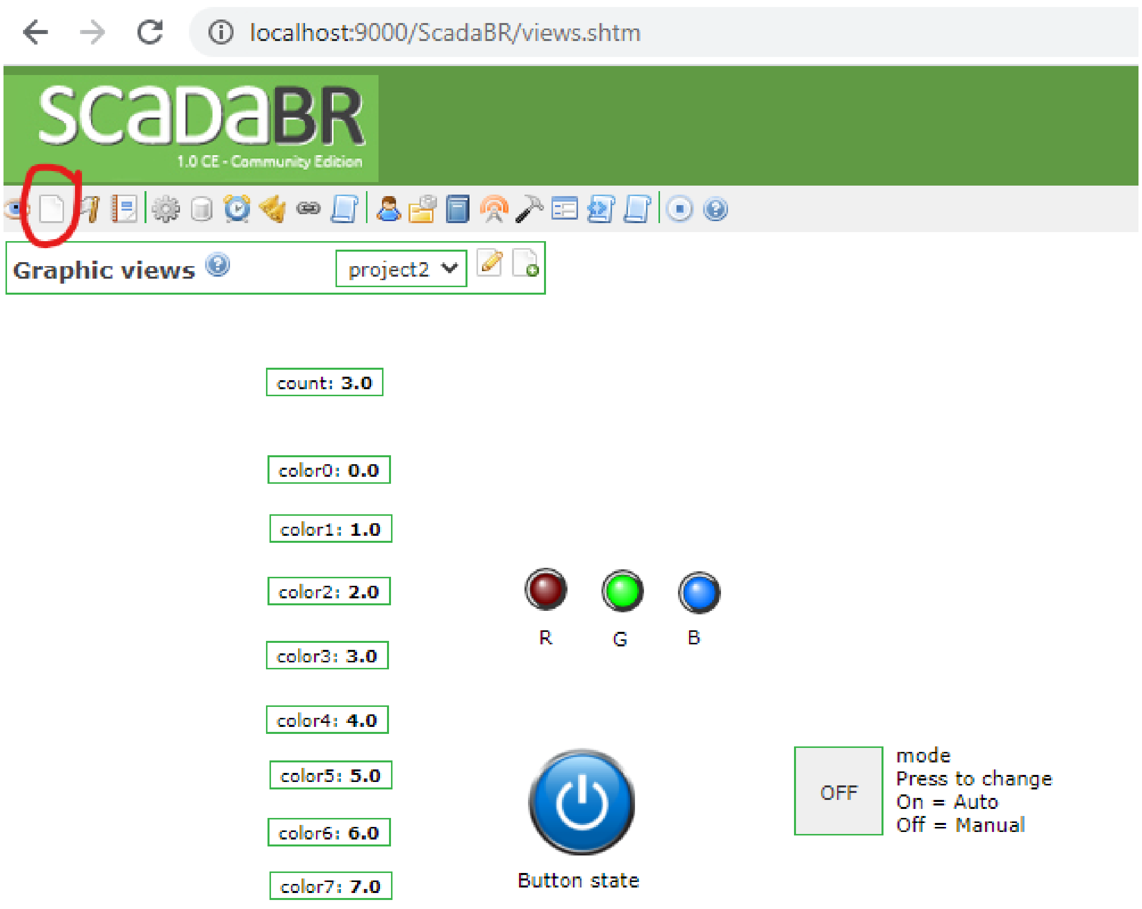
☑ Include point values
☑ Include Uploads folder
☐ Include Graphics folder
Max point values: 100
Select your options and download the project file Export Project

Import Project (Upload)
Select the file (.zip) e click the button to import it
Choose File No file chosen Send

Data

c. Press the Import button marked above.

d. Press the Graphical views button.



e. You need to update the IP address of the host (your PLC) in the data sources screen.
   i. Go to data sources screen.
   ii. On the left side look for Host IP. It is set to 10.0.2.5. Change this to your PLC's IP address.

      To get your PLC's IP address.
      1. Log onto the PLC
      2. Type the command ***ip a***
      3. Record the IP address

f. You may need to stop and restart the Data sources.
   i. Go to the Data sources screen. Press .
   ii. Click the status icon to disable. Press .
   iii. Click the status icon to enable. Press .

6 Post Exercise Report

Answer the following questions.

A. (60 points) Upload a video of your project running. Show the following in your video:
   1. Manual mode operation
      (a) (15 points) show that button presses change the color of the LED
      (b) (15 points) show that HMI count increases after the button presses
   2. Automatic mode operation
      (a) (15 points) show that HMI count loops automatically from 0-7 and restarts.
      (b) (15 points) show that the physical LED color changes approximately every 1 second

B. (20 points) Assume causing the LED to flash RED is a signal to an operator that the system was in an alarm state, how might an attacker cause the RGB LED to flash RED by manipulating the color0-color7 variables and changing the mode.

C. (20 points) If you power off the ScadaBR HMI VM while the PLC program is running does the PLC program and LED circuit continue to operate? Which if any functionality is lost?