PROJECT SPECIFICATION

On-demand Traffic light control

System Design

CRITERIA	MEETS SPECIFICATIONS
Read system requirements	Hardware requirements: 1. ATmega32 microcontroller 2. One push button connected to INT0 pin for pedestrian 3. Three LEDs for cars - Green, Yellow, and Red, connected on port A, pins 0, 1, and 2 4. Three LEDs for pedestrians - Green, Yellow, and Red, connected on port B, pins 0, 1, and 2 Software requirements: In normal mode: 1. Cars' LEDs will be changed every five seconds starting from Green then yellow then red then yellow then Green. 2. The Yellow LED will blink for five seconds before moving to Green or Red LEDs. In pedestrian mode: 1. Change from normal mode to pedestrian mode when the pedestrian button is pressed. 2. If pressed when the cars' Red LED is on, the pedestrian's Green LED and the cars' Red LEDs will be on for five seconds, this means that pedestrians can cross the street while the pedestrian's Green LED is on. 3. If pressed when the cars' Green LED is on or the cars' Yellow LED is blinking, the pedestrian Red LED will be on then both Yellow LEDs start to blink for five seconds, then the cars' Red LED and pedestrian Green LEDs are on for five seconds, this means that pedestrian must wait until the Green LED is on. 4. At the end of the two states, the cars' Red LED will be off and both Yellow LEDs start blinking for 5 seconds and the pedestrian's Green LED is still on. 5. After the five seconds the pedestrian Green LED will be off and both the pedestrian Red LED and the cars' Green LED will be on. 6. Traffic lights signals are going to the normal mode again.
Make full static architecture for your system	1. Define system layers 2. Define system drivers 3. Place each driver into the appropriate layer in the appropriate order 4. Define APIs that will be used for each driver, with its documentation, description, input arguments, output arguments, and return 5. Define the new data types you will use in these drivers Note: You should follow the SOLID principles
Deliver your work	You should deliver, a pdf file containing five sections, system description, system design, system flow chart or state machine, and system constraints if any with a video where you will discuss your work.

Prepare your development environment

CRITERIA	MEETS SPECIFICATIONS
Apply your layered architecture into project's folder structure	1. Create a folder for each layer 2. In each layer folder, create a folder for each driver related to this layer 3. In each driver folder, create .c and .h files 4. Create a main.c file that will call your application"
Prepare all files for development	1. Add header file gaurd to all header files 2. Write all typedefs related to each driver in its header file 3. Write all prototypes for all drivers' APIs in their header files 4. Include lower layer drivers into the .h files of the upper layer/calling drivers 5. Include each driver's .h file into its related .c file 6. Include app.h into main .c
Deliver your work	You should deliver a screenshot of the solution explorer that clarifies your folder structure and a video where you will discuss your work.

Implement your application

CRITERIA	MEETS SPECIFICATIONS
Implement your drivers	"1. Write a skeleton for each function using comments, skeleton means what the function do in simple written steps 1. Convert each step into the appropriate code 2. Each function should return error state to indicate that everything is ok or not"
Test your drivers	"1. Implement a test module for each driver. 1. This test module is simply a main function that call driver's APIs and validate its output 2. These test modules can be manulally test or automated test"
Implement system flow	"1. Revisit system requirements. 1. Write a skeleton in APP_start function into app.c 2. Convert each step into the appropriate code"
Deliver your work	You should deliver your code files, test cases you made on each system driver, and video where you will discuss your work

Test your application

CRITERIA	MEETS SPECIFICATIONS

CRITERIA	MEETS SPECIFICATIONS
User story 1	As a pedestrian when I will make a short press on the crosswalk button while the cars green light is on and pedestrian red light is off, I will wait for the yellow lights to blink for five seconds then the cars red light is on and pededstrian green light is on for five seconds, so that I can cross the street
User story 2	As a pedestrian when I will make a short press on the crosswalk button while the cars yellow light is blinking and pedestrian red light is on, I will wait for all yellow lights to blink for five seconds then the cars red light is on and pededstrian green light is on for five seconds, so that I can cross the street
User story 3	As a pedestrian when I will make a short press on the crosswalk button while the cars red light is on and pedestrian green light is on, I expect nothing to be done
User story 4	As a pedestrian when I made a long press on the crosswalk button, I expect nothing to be done.
User story 5	As a pedestrian when I made a double press on the crosswalk button, I expect that the first press will do the action and nothing to be done after the second press.
Deliver your work	7- You should deliver test cases results and video where you show executing all the test cases using one of the simulators (Proteus if you have licence or using ATMEL STUDIO) or by hardware if applicable