Lab Sheet — Integrated Peripheral Application

Smart Traffic Light Controller with Pedestrian Crossing

Student Name:	Score:	

1. Objective

In this lab, students will design and simulate a **Smart Traffic Light Controller** using the ATmega32 microcontroller and CodeVisionAVR. The system integrates four peripheral types — LEDs, push buttons, a 7-segment display, and a 16×2 LCD — into one real-world embedded application.

2. System Description

The system simulates a traffic light at a pedestrian crossing. It operates in three states:

State	Active LED	LCD Message	7-Segment	Duration
NORMAL	**GREEN**	Line 1: "Traffic: GO" Line 2: "Press BTN to cross"	— (off)	Indefinite
WARNING	**YELLOW**	Line 1: "Please Wait..." Line 2: "Crossing soon"	Counts 3→1	3 seconds
CROSSING	**RED**	Line 1: "Safe to Cross!" Line 2: "Time left: X sec"	Counts 9→0	10 seconds

When the **push button** is pressed during the NORMAL state, the system transitions to WARNING, then CROSSING, and finally returns to NORMAL automatically.

3. Hardware Components & Pin Assignment

Components	ATmega32 Port - Pin	
Green LED	PORTA.0	Traffic GO indicator
Yellow LED	PORTA.1	Warning indicator
Red LED	PORTA.2	Stop / pedestrian crossing
Push Button	PORTB.0	Active LOW — use internal pull-up (set PORTB.0 = 1)
7-Seg (a–g)	PORTC.0 – PORTC.6	Common Cathode; segment a→PC0 … segment g→PC6
LCD (4-bit)	PORTD	RS=PD0, EN=PD1, D4–D7=PD4–PD7 (alcd.h default)

Note: All LEDs should be connected with a 220Ω current-limiting resistor to GND.

4. 7-Segment Display — Digit Encoding (Common Cathode)

The table below gives the required PORTC value to display each digit (segments a–g on PC0–PC6):

0	1	2	3	4	5	6	7	8	9
0x3F	0x06	0x5B	0x4F	0x66	0x6D	0x7D	0x07	0x7F	0x6F

Circuit Diagram

Attach or draw your Proteus circuit diagram in the space below. Label all component connections clearly.

[Attach / Draw Circuit Diagram Here]

HINT :

- Build the circuit in Proteus using ATmega32, 3 LEDs (green/yellow/red), a push button with pull-up resistor, a common-cathode 7-segment display, and a 16×2 LCD.
- Verify all connections match the pin assignment table in Section 3.

Complete the Code

- Ensure the state machine transitions: NORMAL → WARNING → CROSSING → NORMAL.
- The LCD must update correctly in every state.
- The 7-segment must count down in WARNING (3→1) and CROSSING (9→0).

Simulation & Verification

- Run the simulation and demonstrate all three states to the instructor.
- Press the button multiple times; the system must behave correctly each cycle.