Embedded Indicator Control System Architecture

Project Overview

This project demonstrates an Embedded Indicator Control System for vehicles using **Embedded C** (MSYS2 with GCC compiler) and **MATLAB Simulink** for simulation. The goal is to implement left, right, and hazard indicators with GPIO logic, pulse generation, and UART logging.

Tools and Technologies Used

- **C Language** (Embedded C)
- **MSYS2 Terminal** (UCRT64, GCC Compiler)
- **MATLAB Simulink (R2025a)**
- **UART Logging** (via `printf`)
- **Simulink Scopes** for waveform monitoring

Application Code Structure

1. `main.c`

- Contains the main application logic.
- Initializes UART and sends commands to simulate indicator behavior.
- Simulates hazard mode activation and deactivation with toggling logs.

2. `uart_driver.c` and `uart_driver.h`

- Contains UART utility functions:
  - `UART_Init()` — Simulates UART initialization
  - `UART_SendString()` — Sends messages to terminal
  - `UART_LogStatus()` — Logs current system status

Example Output in Terminal:

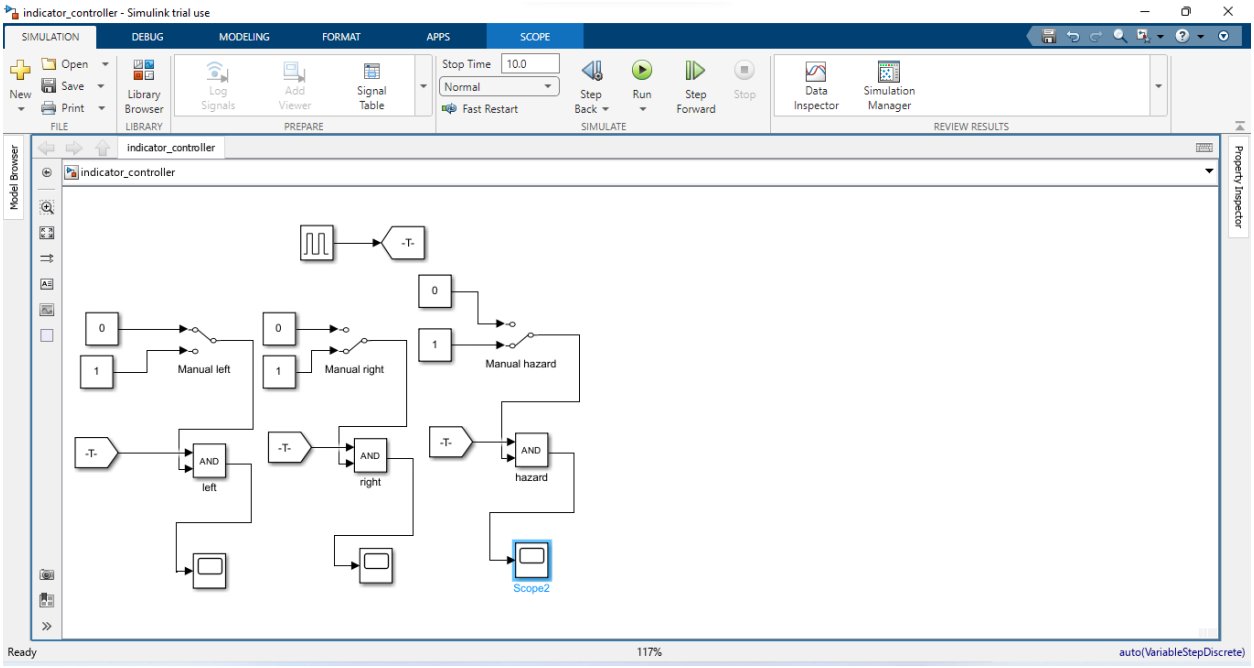
```
UART Initialized.
Status: Hazard Mode Activated
Toggling Hazard Light (Left + Right)...
...
Status: Hazard Mode Deactivated
```

Simulink Model Description

Simulink Components Used:

- **Pulse Generator** – Generates square waveform for blinking effect
- **Manual Switches** – Used for Left, Right, Hazard control
- **AND Gates** – Controls logic for each indicator based on switches
- **Scopes** – Used to visualize blinking pattern (ON/OFF)

Simulink Diagram:



Each AND gate output goes to a separate Scope block:

- Scope 1: Left Indicator
- Scope 2: Right Indicator
- Scope 3: Hazard (Left + Right)

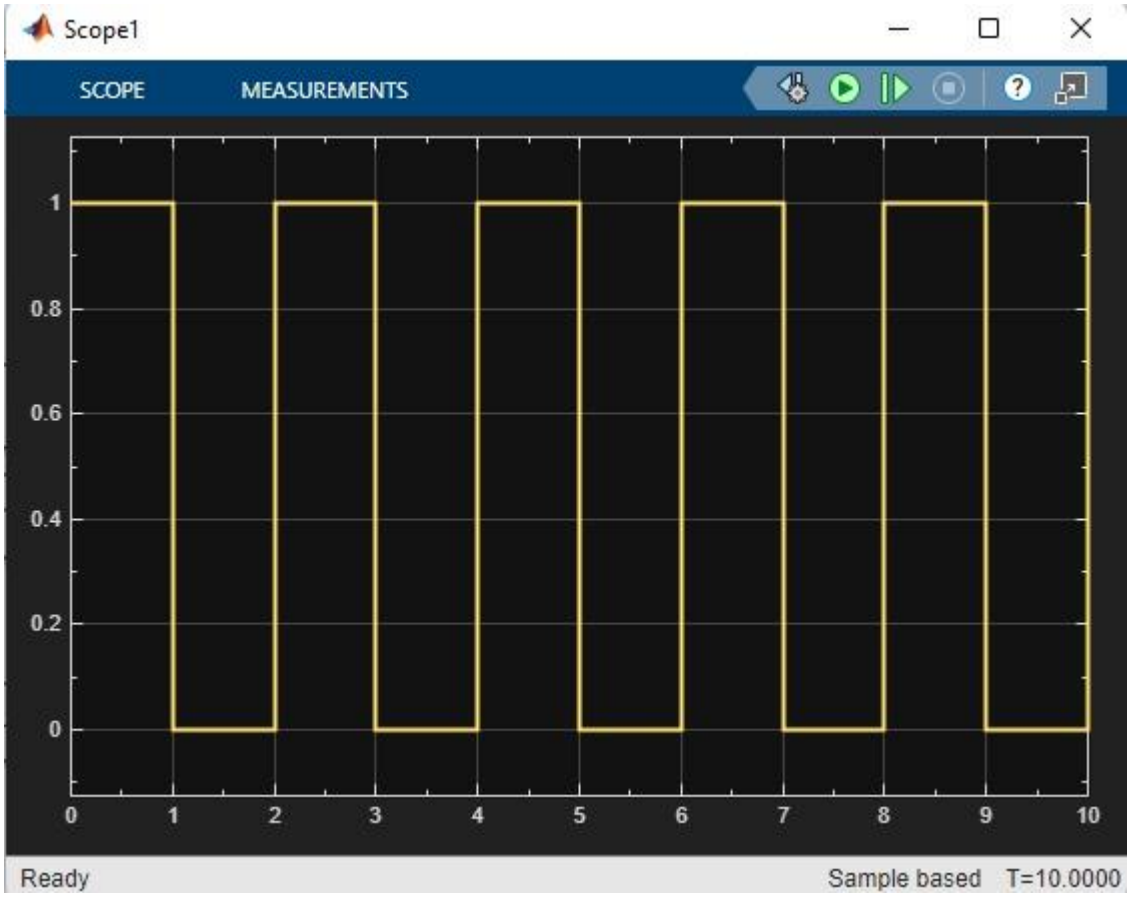
Output Results

1. MSYS2 UART Output:

```
UART Initialized.
Status: Hazard Mode Activated
Toggling Hazard Light (Left + Right)...
...
Status: Hazard Mode Deactivated
```

2. Simulink Output Graph:

- Square waveform observed from Pulse Generator
- Waveform toggles between 0 and 1 at defined interval (period)
- Switching between 0 and 1 shows ON/OFF states



Conclusion

This project integrates simulation and embedded programming concepts:

- UART simulation using `printf`
- Basic logic control using Simulink blocks
- Visual indicator behavior using scopes

It demonstrates how embedded systems can use both hardware-level code and high-level simulation to model real-time behavior efficiently.

Prepared by: Jangam Sindhuja

Date: 18th July 2025