“All scenariosCAR PARKING SYSTEM

VHDL SIMULATION LAB REPORT

1-) Title And Authors

Lab 1:Car Parking System (VHDL Simulation)

Course: Fundamentals of Electronics / Logic Networks

Authors:

- **Francesco Bertaiola**
- **Stefano Giroletti**
- **Bekir Hamza Gümüş**

2-) Objectives and Specifications

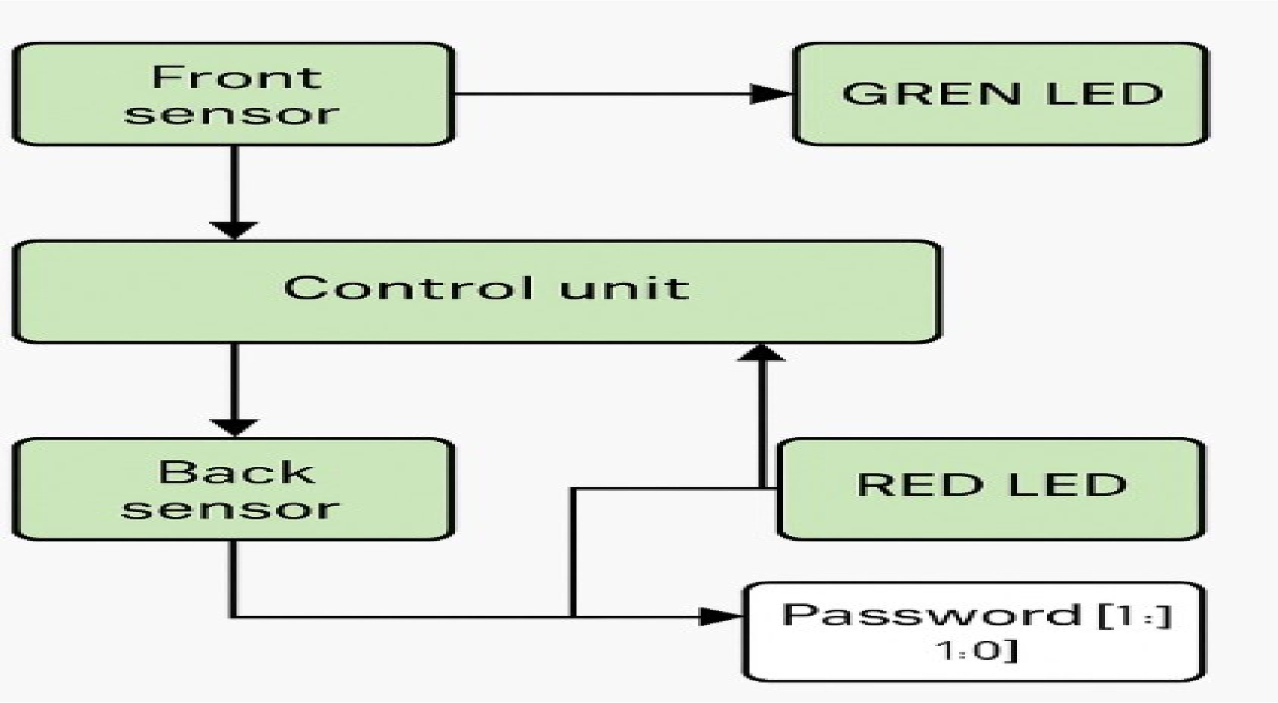
The objective of this laboratory activity was to design, implement, and simulate a Car Parking Control System using VHDL and GHDL + GTKWave.
The system detects vehicle arrivals through sensors, processes password inputs, manages multiple cars, and displays appropriate messages on a 7-segment display.

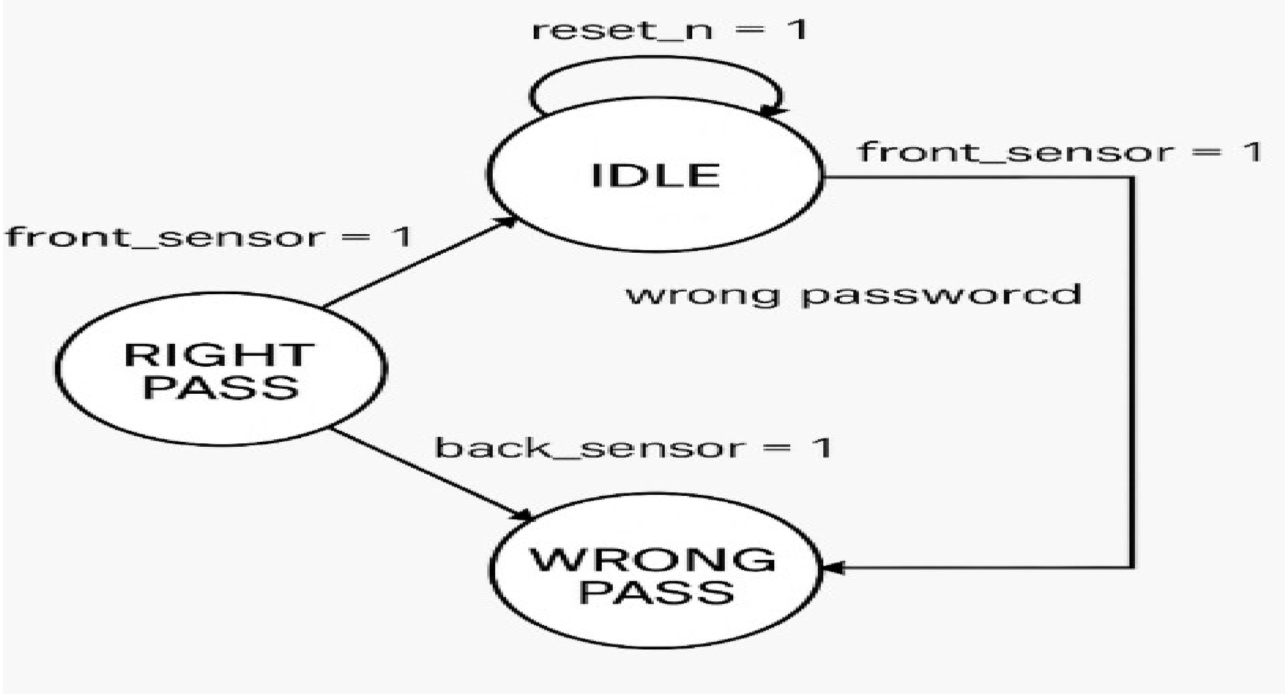
The system must meet the following specifications:

- Asynchronous, active-low reset (reset_n = '0')
- Two digit passwords (password_1, password_2)
- Two entrance sensors: front_sensor, back_sensor
- Three LED indicators:
  A. Green LED → Successful entry
  B. Red LED → Wrong password or STOP condition
- 7-segment outputs:
  1. “En” → Wait for password
  2. “EE” → Wrong password
  3. “GO” → Correct password
  4. “SP” → STOP (Multiple cars)
- STOP mode is triggered when car 2 arrives before car 1 leaves.

The system must return to IDLE after each operation

3-) Block Diagram





Main components:

- FSM (Finite State Machine) implementing IDLE, WAIT_PASSWORD, RIGHT_PASS, WRONG_PASS, and STOP
- Password Comparator
- Counter (10-cycle waiting time)
- 7-segment Encoder
- LED Controller

4-) **Components Description**

4.1 Finite State Machine (FSM)

Implements 5 states:

- IDLE
- WAIT_PASSWORD
- RIGHT_PASS
- WRONG_PASS
- STOP

Transitions depend on:

- Sensor inputs
- Password correctness
- Counter expiration

4.2 Counter

Counts 10 clock cycles during password waiting phase.
If the counter reaches 9, the system evaluates the password.

4.3 Password Comparator

Compares password_1 and password_2 with the expected value "01" / "10".

4.4 7-Segment Encoder

Converts symbolic outputs into 7-segment codes:

- E → 0000110
- N → 0101011
- G → 0000010
- O → 1000000
- S → 0100100
- P → 0001100
- All off → 1111111

4.5 LED Driver

- Green LED blinks during RIGHT_PASS
- Red LED blinks during WRONG_PASS and STOP

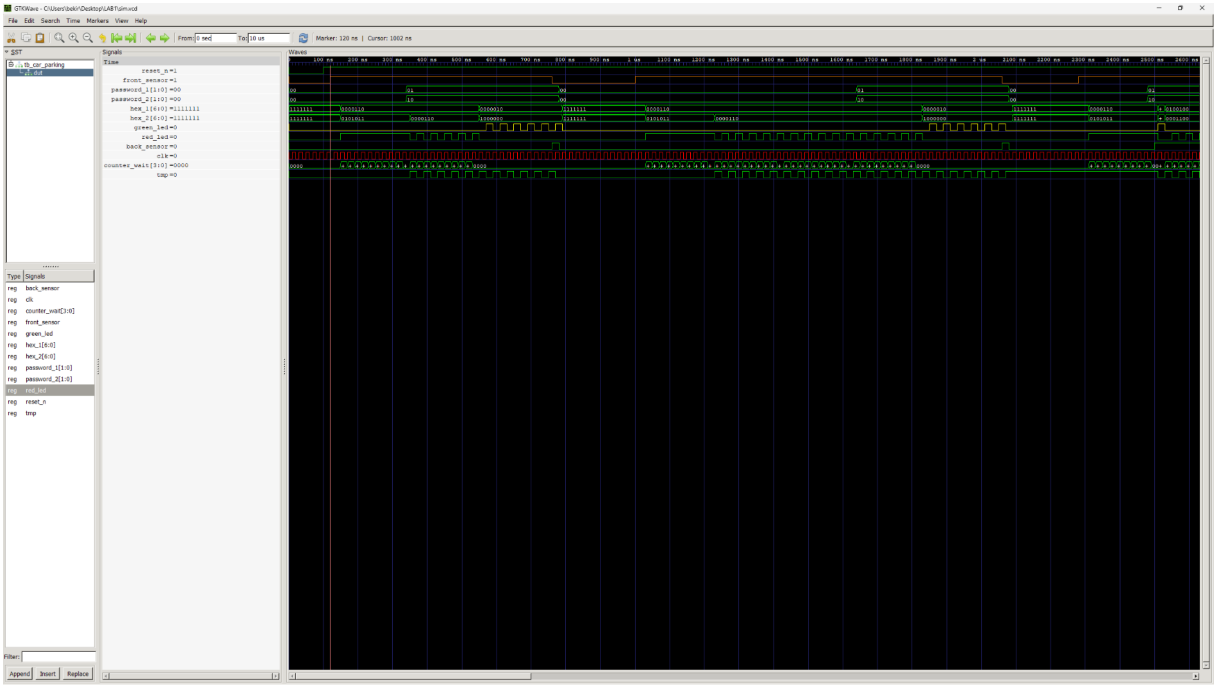
5-) Simulation and Results

Simulations were performed using GHDL and visualized in GTKWave.
All scenarios required in the laboratory instructions were successfully reproduced.

5.1 Scenario 1 — Normal Entry

Description:
A car arrives (front_sensor='1').
After a 10-cycle waiting time, a correct password ("01", "10") is provided.
The system displays "GO", green LED blinks, and after back_sensor='1', the system returns to IDLE.

Waveform Evidence:



GTKWave - C:\Users\bekir\Desktop\LAB1\sim.vcd

From: 0 sec To: 10 us

Marker: 120 ns | Cursor: 1002 ns

```
reset_n =1
front_sensor =1
password_1[1:0] =00
password_2[1:0] =00
hex_1[6:0] =1111111
hex_2[6:0] =1111111
green_led =0
red_led =0
back_sensor =0
clk =0
counter_wait[3:0] =0000
tmp =0
```

Type	Signals
reg	back_sensor
reg	clk
reg	counter_wait[3:0]
reg	front_sensor
reg	green_led
reg	hex_1[6:0]
reg	hex_2[6:0]
reg	password_1[1:0]
reg	password_2[1:0]
reg	red_led
reg	reset_n
reg	tmp

Key observations:

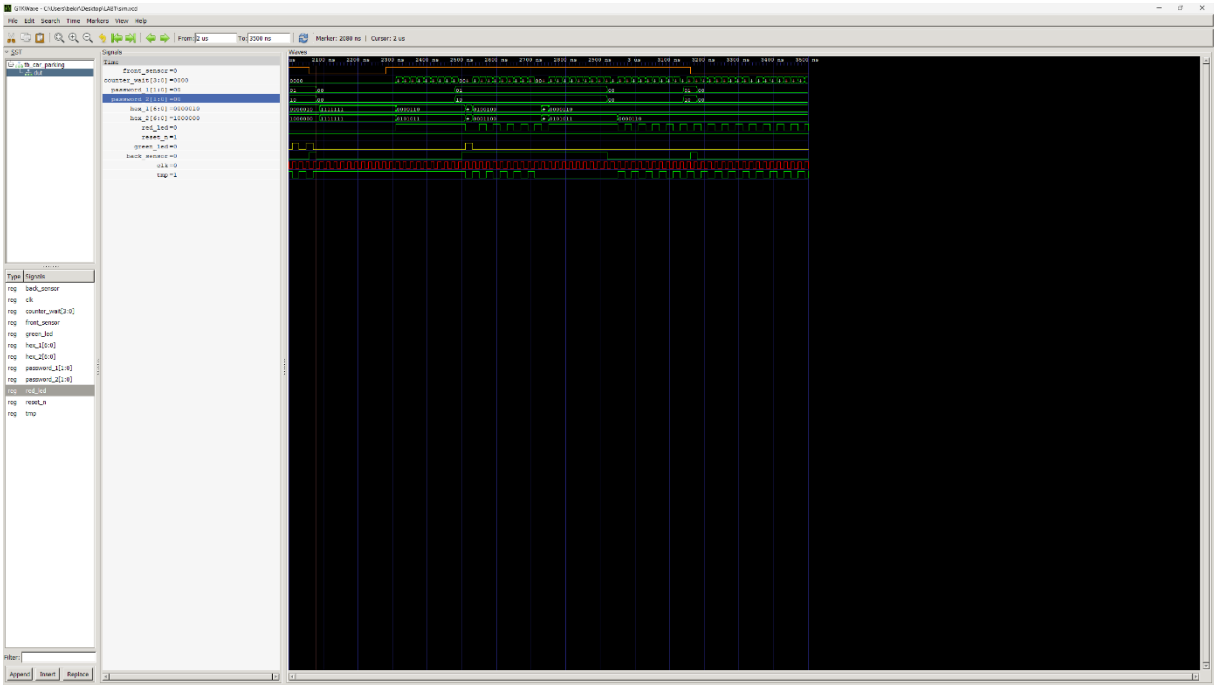
- front_sensor='1' → FSM enters WAIT_PASSWORD
- Counter counts from 0 to 9
- Correct password triggers RIGHT_PASS
- hex_1="0000010", hex_2="1000000" → "GO"
- System returns to IDLE when back_sensor='1'.

5.2 Scenario 2 — Wrong Password

Description:

A car arrives but enters an incorrect password ("00", "00").
The system enters WRONG_PASS, displays “EE”, and the red LED blinks.
A correct password is then entered to exit the error state.

Waveform Evidence:



GTKWave - C:\Users\bekir\Desktop\LAB7\sim.vcd

From: 2 us To: 3500 ns

Marker: 2000 ns | Cursor: 2 us

```
front_sensor =0
counter_wait[3:0] =0000
password_1[1:0] =00
password_2[1:0] =00
hex_1[6:0] =0000010
hex_2[6:0] =1000000
red_led =0
reset_n =1
green_led =0
back_sensor =0
clk =0
tmp =1
```

Key observations:

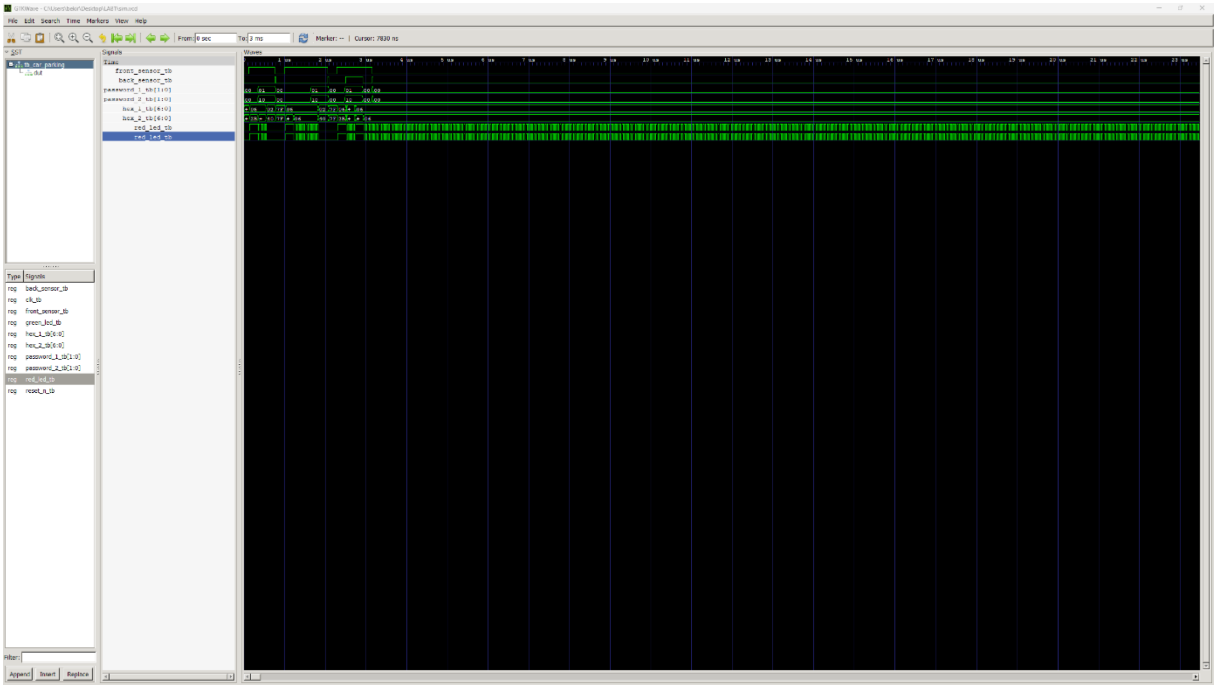
- During WAIT_PASSWORD, wrong password applied
- Display shows “EE”
- red_led toggles
- Correct password transitions system to RIGHT_PASS

5.3 Scenario 3 — Multiple Cars (STOP State)

Description:

Car 1 enters with correct password (RIGHT_PASS).
Before Car 1 leaves (back_sensor='1'), Car 2 arrives (front_sensor='1').
The system enters STOP, displaying “SP”, and blinking the red LED.

Waveform Evidence:



Key observations:

- Car 1 RIGHT_PASS → GO
- front_sensor='1' and back_sensor='1' **simultaneously**
- FSM enters **STOP**
- hex_1="0100100", hex_2="0001100" → “SP”
- After Car 1 leaves (back_sensor='0'), Car 2 can enter normally

5.4 Scenario 4 — Reset

Description:
When reset_n='0' is asserted, the system must immediately return to IDLE regardless of its current state.

Observations:

- All outputs reset to default
- Display turns off (1111111)
- LED outputs go low
- FSM re-enters IDLE

6-) Credits

- VHDL inspired by lecture materials
- GTKWave screenshots generated by the group
- External diagrams/tools referenced: Antares