CmpE 443 Final Project Design Document
Group Name : R3S3RV3D

Members
Dilruba Reyyan KILIÇ (Team Leader)
Yusuf KALAYCI
Bekir Burak ASLAN
Mustafa Enes ÇAKIR

November 20, 2018
Version 1.00

Contents

1 Introduction

In this report, we will introduce the progress of our group: R3S3RV3D on the final project of CmpE 443 (Embedded Systems) Course.

At this point, the Car includes 4 LED's, 1 MotorController and 1 Speed Sensor in addition to the Main Driver LPC4088. The car is able to move forward, backwards and turn left or right. The actions and the conditions are explained below.

The challenge is to bring these different devices and controllers together without having any errors or any other complications and create the desired car of the project (Figure 1). It is both challenging at software and hardware design.

In order to explain the hardware connections and software-hardware relations, we prepared this report.

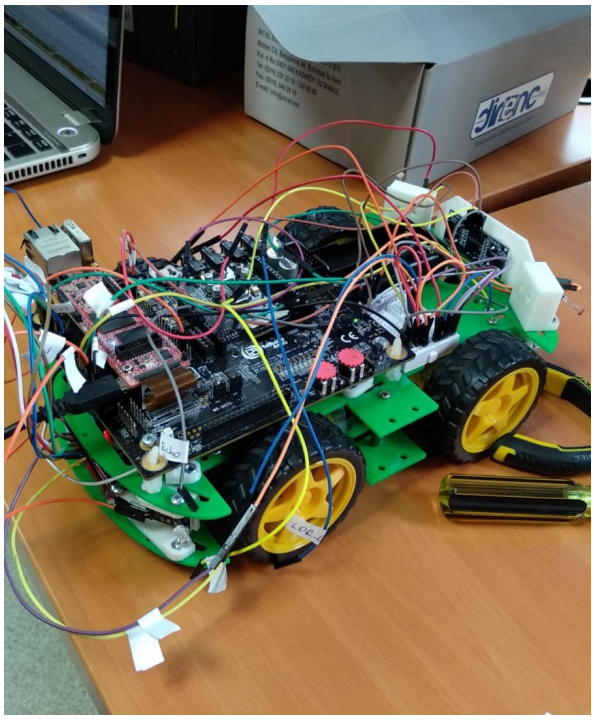

Figure 1: R3S3RV3D Car

2 System Level Structural Diagram (Block Diagram)

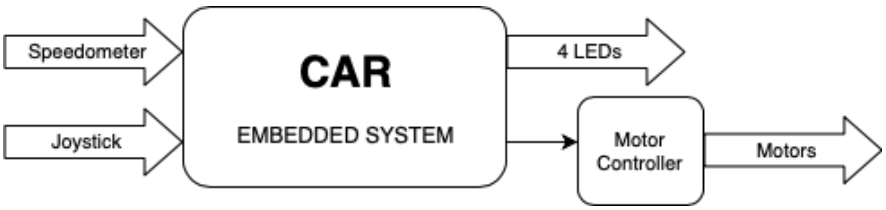

Figure 2: System Level Structural Diagram

2.1 Inputs

2.1.1 Speedometer

Speedometer counts rotation number of the wheels. In order to obtain a 90 degrees turn of the car to the left or right, Speedometer counts the holes on the wheels and determine the necessary rotation count.

2.1.2 Joystick

The actions of the car is controlled via the joystick. We can list these actions as:

- When **Joystick Left** button is pressed, the car rotates 90 degree in counter-clockwise direction. The car stops after the rotation completes.
- When **Joystick Right** button is pressed, the car rotates 90 degree in clockwise direction. The car stops after the rotation completes.
- When **Joystick Up** button is pressed, the car goes in forward direction.
- When **Joystick Down** button is pressed, the car goes in backward direction.
- When **Joystick Center** button is pressed, the car stops.

The LEDS actions are explained in the LEDs section.

2.2 Outputs

2.2.1 LEDs

The LEDs represents the direction or action of the car. The following states of the LED is changed according to the action which the car performs:

- All LEDs are **OFF**, while the car is not moving.
- The front LEDs are **ON**, while the car is moving forward.
- The back LEDs are **ON**, while the car is moving backward.
- The left LEDs are **BLINKING**, while the car is turning left.
- The right LEDs are **BLINKING**, while the car is turning right.

2.2.2 Motor Controller

The Motor Controller provides the connection between the main driver (LPC4088) and the motors (wheels). The first channel of the Motor Controller controls the left wheels and the second channel of the Motor Controller controls the right wheels. We can obtain the following actions as:

- The car moves FORWARD when IN1 & IN3 are set to 1, and IN2 & IN4 are set to 0.
- The car moves BACKWARD when IN2 & IN4 are set to 1, and IN1 & IN3 are set to 0.
- The car STOPS when IN1 & IN2 & IN3 & IN4 are set to 0.
- The car TURNS RIGHT when IN2 & IN3 are set to 1, and IN1 & IN4 are set to 0.
- The car TURNS LEFT when IN1 & IN4 are set to 1, and IN2 & IN3 are set to 0.

3 Sequence Diagram

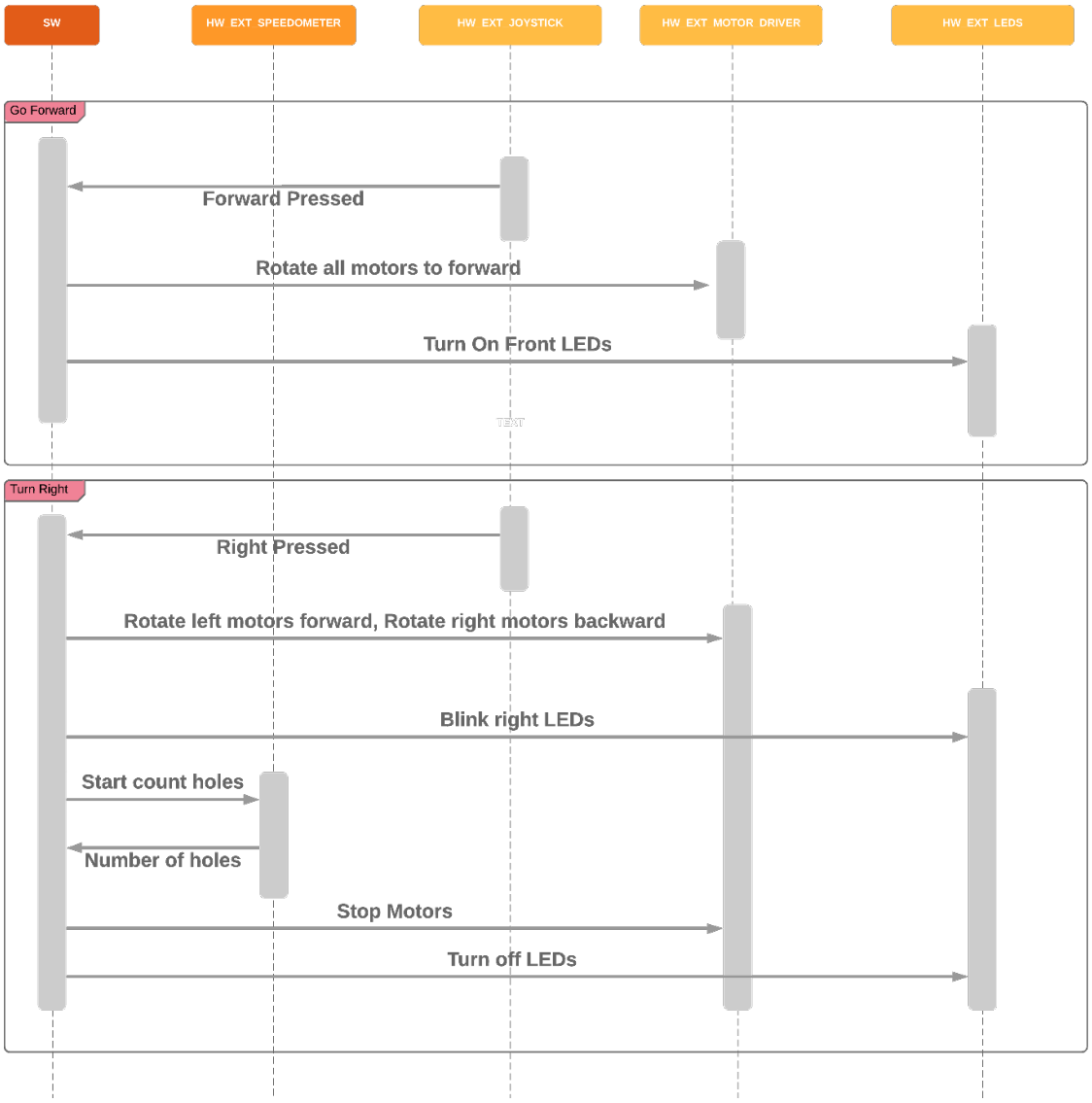

Figure 3: Sequence Diagram

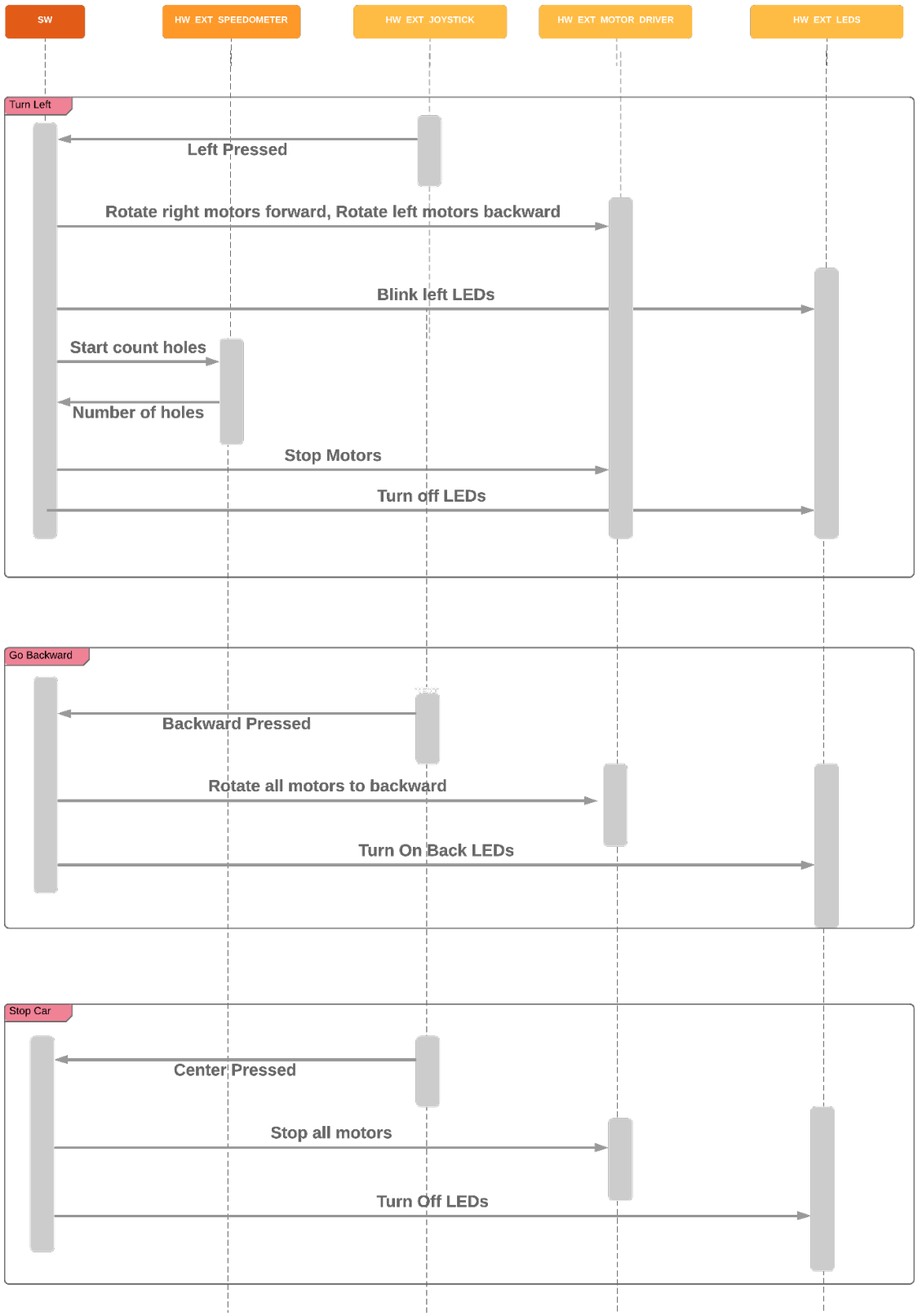

Figure 4: Sequence Diagram

4 LED Connections

The LED Name	LPC4088 Pin	Pin Functionality	Reason
Front-left	P0.26 (P18)	GPIO	It is a GPIO pin which is not conflicting with any PWM or Timer functionalities. Also, it controls ON/OFF state of LED.
Front-right	P0.25 (P17)	GPIO	It is a GPIO pin which is not conflicting with any PWM or Timer functionalities. Also, it controls ON/OFF state of LED.
Back-left	P1.31 (P20)	GPIO	It is a GPIO pin which is not conflicting with any PWM or Timer functionalities. Also, it controls ON/OFF state of LED.
Back-right	P1.30 (P19)	GPIO	It is a GPIO pin which is not conflicting with any PWM or Timer functionalities. Also, it controls ON/OFF state of LED.
Front-left, Front-right, Back-left, Back-right	P1.20 (P7)	PWM1[2]	It has PWM functionality. It provides BLINKING LEDs.

Table 1: Selected LED pins

Note: The + (positive) poles of the LED's are connected to GPIO's and the - (negative) poles are connected to a PWM connection on the breadboard instead of GND. We did not need 4 different PWM's for every single LED since they should blink as the same, instead we connected all the LED's in one PWM connection and controlled the ON & OFF situations from their related GPIO pins.
By this, we achieved a simpler design and the LED's blink harmoniously. You can see the circuit level design from Figure 5.

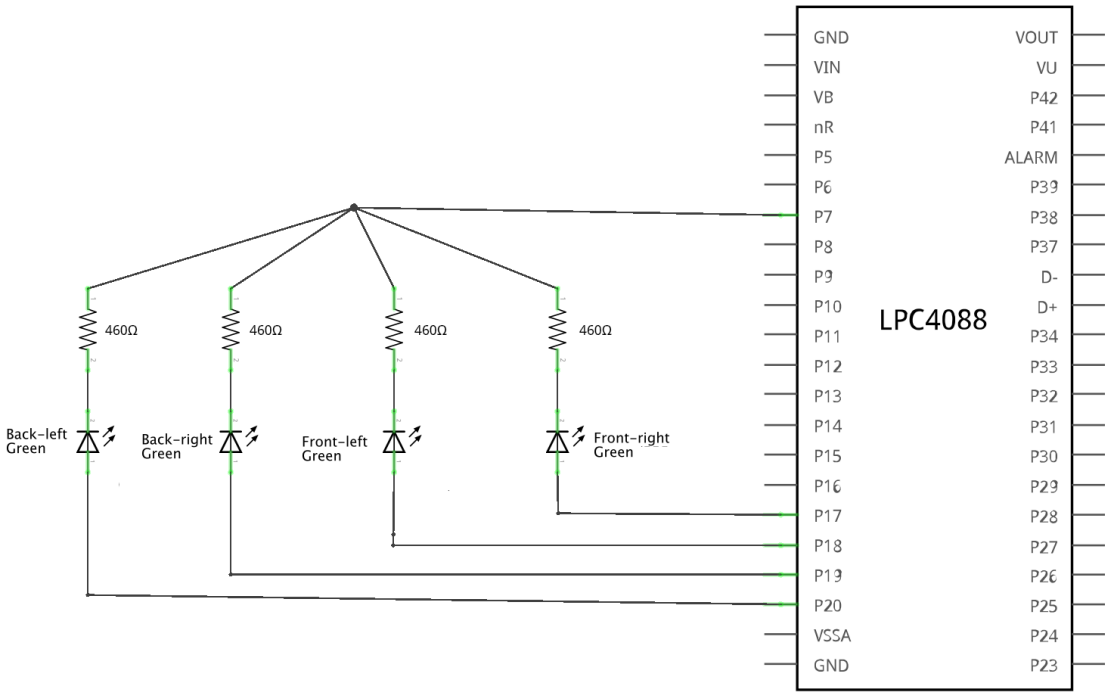

Figure 5: LEDs Circuit Diagram

5 Motor Speed Sensor Connection

The Speed Sensor Name	**LPC4088 Pin**	**Pin Functionality**	**Reason**
Left	P0.23	T3_CAP_0	We need a timer to use the speedometer, since there are two Timer3 capture pins we chose CAP_0 pin. We capture falling/rising edges.
Right	-	-	We don't use it for first interim

Table 2: Selected Speedometer pins

The Speedometer is connected to the LPC4088 and controls the Motors from the Software. After Joystick Left or Right button is pressed, the Speedometer starts to count the holes on the wheels and stops the motors from the software when the necessary number is reached.

6 Motor - Driver Connection

Motor Terminal	**Motor Driver Terminal**
Motor FL +	Output A +
Motor FL -	Output A -
Motor BL +	Output A +
Motor BL -	Output A -
Motor FR +	Output B +
Motor FR -	Output B -
Motor BR +	Output B +
Motor BR -	Output B -

Table 3: Motor - Driver Connection

A 2-Channel Motor Controller is used to control 4 Motors. Since the right wheels and the left wheels needs to move simultaneously between theirselves, the BL & FL motors are connected to the same ports and the BR & FR motors are connected to the same ports on the Motor Controller.

7 Driver - Board Connection

Motor Driver Pin Name	**LPC4088 Pin**	**Pin Functionality**	**Reason**
ENA	P1.11 (P25)	PWM0[6]	Since we chose PWM1 for LEDs in order to prevent any conflicts, we chose PWM0 for motor.
ENB	P1.11 (P25)	PWM0[6]	All motors should rotate at the same speed, so connected it to the same pin.
IN1	P1.23 (P6)	GPIO	It is a GPIO pin which is not conflicting with any PWM or Timer functionalities. It is used to control channel A motors' direction.
IN2	P1.24 (P5)	GPIO	It is a GPIO pin which is not conflicting with any PWM or Timer functionalities. It is used to control channel A motors' direction.
IN3	P0.21 (P8)	GPIO	It is a GPIO pin which is not conflicting with any PWM or Timer functionalities. It is used to control channel A motors' direction.
IN4	P0.1 (P10)	GPIO	It is a GPIO pin which is not conflicting with any PWM or Timer functionalities. It is used to control channel A motors' direction.

Table 4: Driver - Board Connection

8 Conclusion

The R3S3RV3D Car is working properly. All the functions and requirements on the Final Interim Project are complete.
With this project we achieved skills like:

- Determining the correct pins on a Microcontroller Board with the desired functionality.
- Contolling multiple LEDs.
- Controlling and Installing multiple DC motors.
- Getting feedbacks for DC motors and acting accordingly.
- Using the Speedometer for different needs.
- Using a Timer in a complicated system.
- Using the PWM channel in different needs. (Motor & LED's)
- Working as a team.
- Designing the BreadBoard.
- Learning that jumper cables are hard to find and the best way to get one is to go to Karaköy :)