CmpE 443 Final Project Design Document
Group Name: Tezla

Members
Abdurrahman DILMAC (Team Leader)
Ahmet Semih ARI
Ramazan ARSLAN
Yunus Emre DEMIRCI

December 29, 2018
Version 2.00

Contents

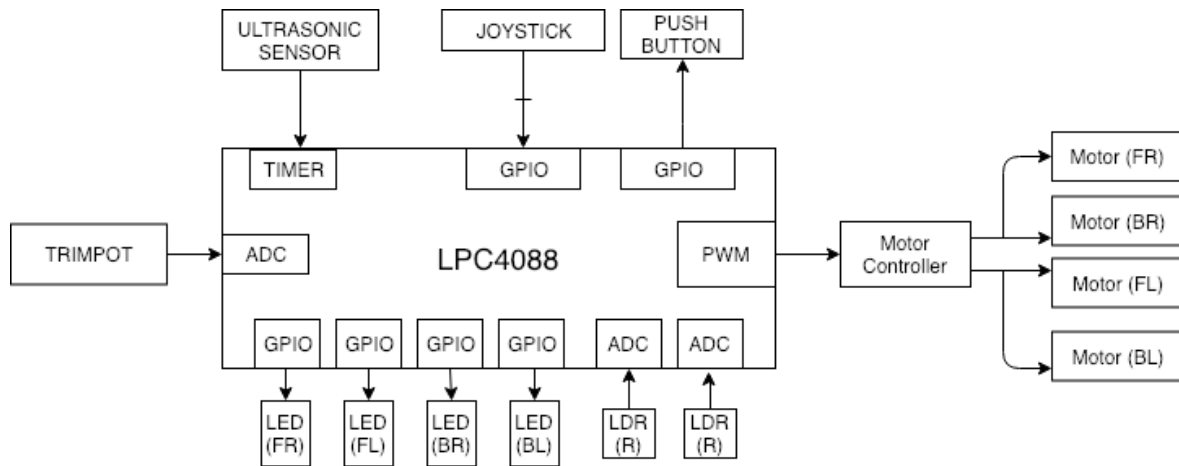# 1 System Level Structural Diagram (Block Diagram)

Figure 1: Block Diagram of car.

LPC4088 Microcontroller: Main part of the car. Runs software and controls peripherals.

Motor controller: By getting inputs from the main board, controls motors by means of controlling speed, directions, soft and hard brakes.

Motor: Gives tork to the car, and thus moves.

LEDs: Emits light, controlled by the main board. Used for car signals.

Joystick: Takes input from the player/user and informs the main board, so it can take appropriate action.

Push Button: It is used to toggle mode to Manuel and Auto with interrupts.

Ultrasonic sensor: Car can measure distance with ultrasonic sensor.

Trimpot: Used to adjust motor speed. It can take value between 0-100.
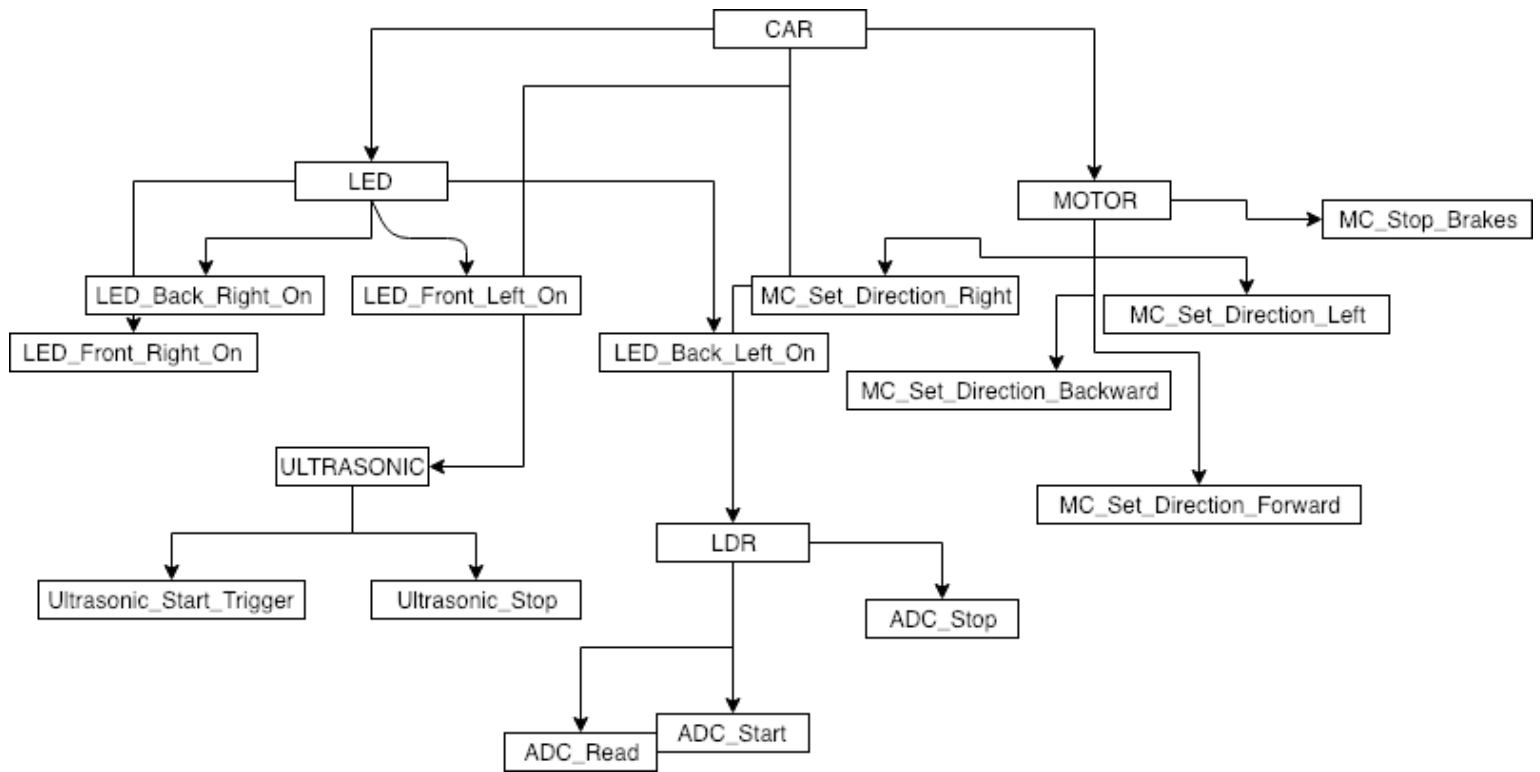
2 System Level Functional Diagram

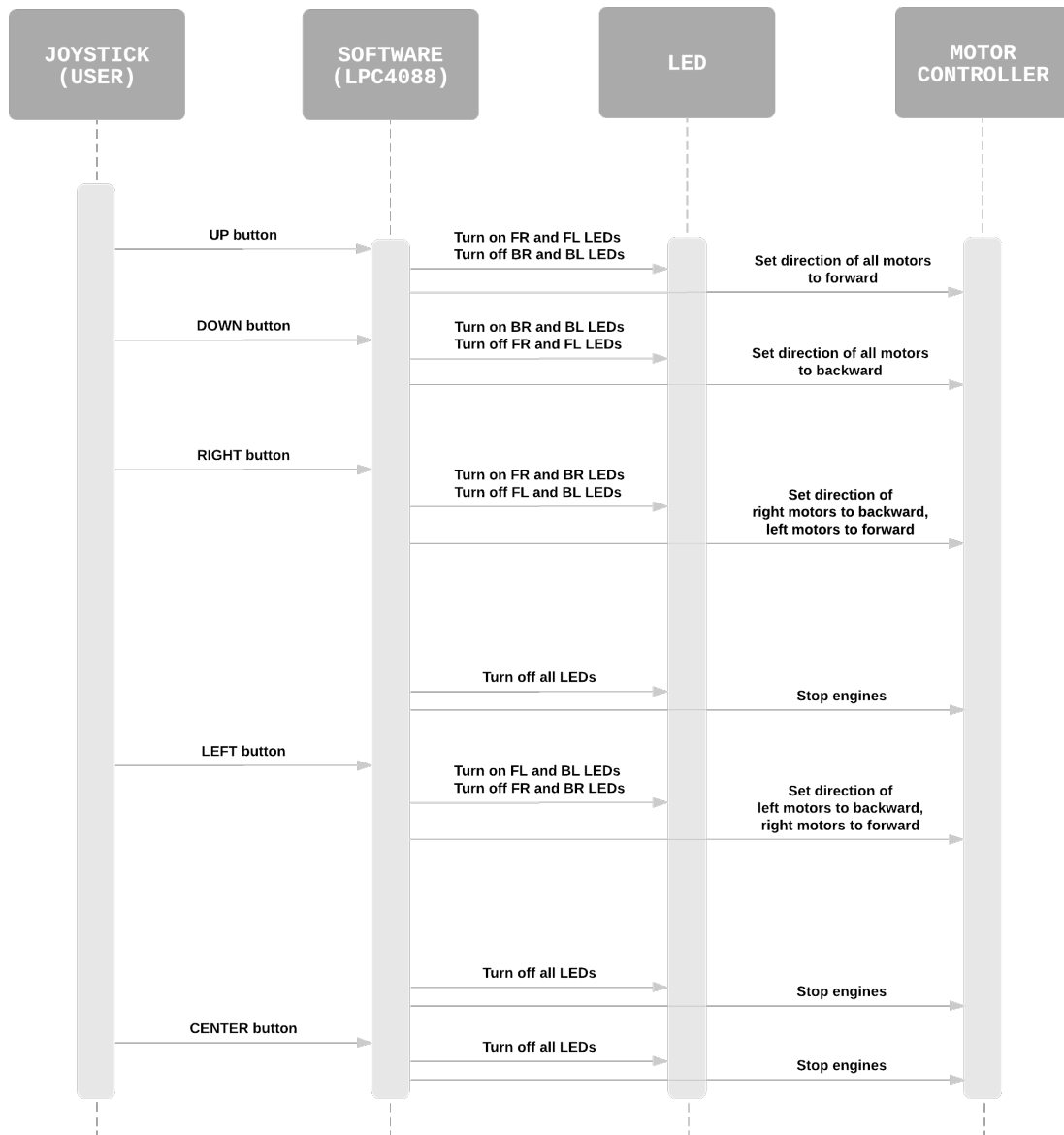Figure 2: System Level Functional Diagram

3 Sequence Diagrams

3.1 Manual Mode

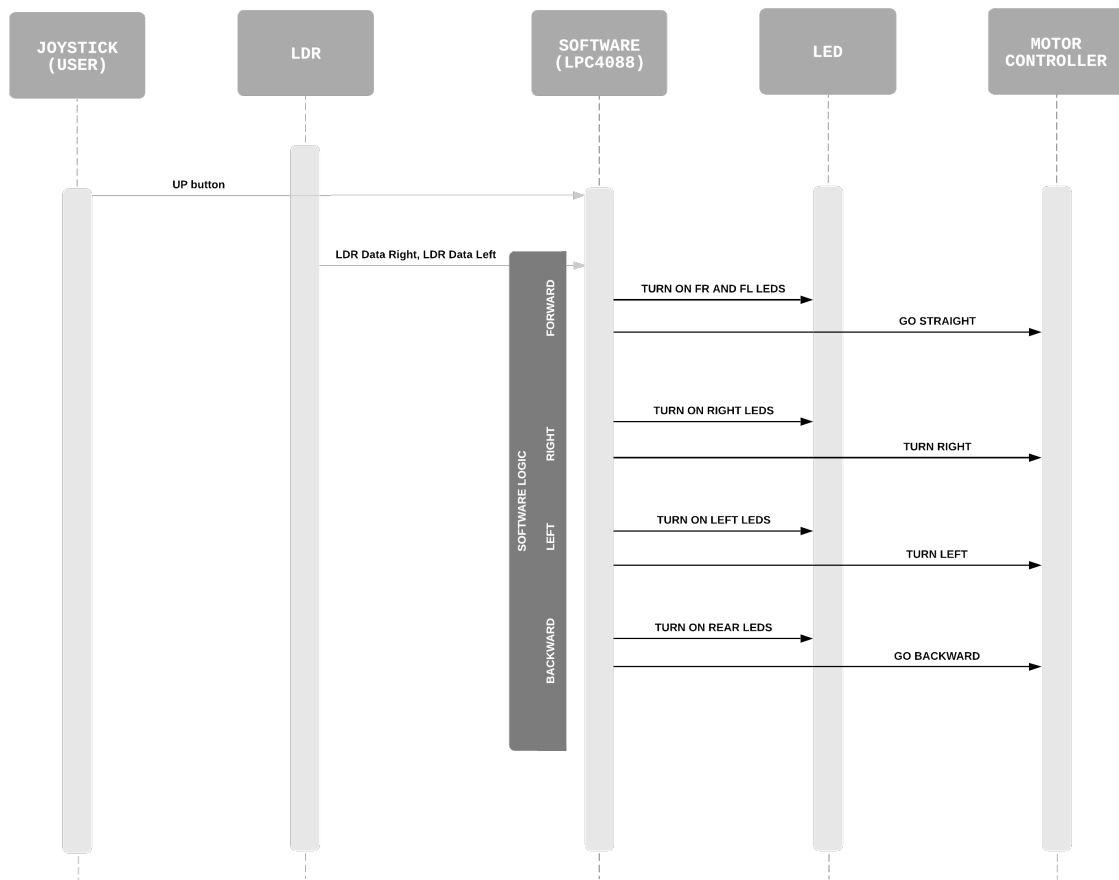Figure 3: Sequence Diagram of Manual Mode.

3.2   Autonomous Mode

Figure 4: Sequence Diagram of Autonomous Mode.

4 Connections

4.1 LED Connections

All the components which are controlled via LPC4088 should be connected to the board. Therefore, you should determine the pins and their functionalities. Draw the described table in the term project interim report description document. After you determine the all the pins, draw the circuit schematic for the LED circuits.

The LED name	LPC4088 PIN	Port	Pin Functionality	Reason
Front Right LED	PIN5	P1.24	GPIO	Supports digital I/O
Front Left LED	PIN6	P1.23	GPIO	Supports digital I/O
Back Right LED	PIN7	P1.20	GPIO	Supports digital I/O
Back Left LED	PIN8	P0.21	GPIO	Supports digital I/O

Table 1: LED Pins and their functionalities

4.2 LDR Connections

Motor driver PIN	LPC4088 PIN	Port	Pin Functionality	Reason
GND	PIN1	GND	Ground	Common ground
LDR left	PIN17	PORT0_25	ADC0_IN[2]	Convert digital
LDR right	PIN18	PORT0_26	ADC0_IN[3]	Convert digital
VCC	PIN45	Vout	Power input	3.3V needed

Table 2: Ultrasonic Sensor and Board PIN connections with their functionalities.

4.3 Motor - Motor Controller Connections

There is only one motor controller to control 4 motors. Output A is used for right motors and Output B is for left motors.

Motor Terminal	Motor Driver Terminal
Motor 1 +	Output A +
Motor 1 -	Output A -
Motor 2 +	Output A +
Motor 2 -	Output A -
Motor 3 +	Output B +
Motor 3 -	Output B -
Motor 4 +	Output B +
Motor 4 -	Output B -

Table 3: Motor Controller and Motor connections.

4.4 Motor Controller - Board Connections

Here is the table depicting relations between the board and motor controller:

Motor driver PIN	LPC4088 PIN	Port	Pin Functionality	Reason
Power GND	PIN1	GND	Ground	Ground is needed
+5V Power	PIN44	Vout	Power input	For transferring voltage
A Enable	PIN29	P1.3	PWM0_1 (ENA)	Supports PWM Output
B Enable	PIN30	P1.2	PWM0_2 (ENB)	Supports PWM Output
IN1	PIN12	P0.8	GPIO	Supports digital I/O
IN2	PIN13	P0.7	GPIO	Supports digital I/O
IN3	PIN9	P0.0	GPIO	Supports digital I/O
IN4	PIN10	P0.1	GPIO	Supports digital I/O

Table 4: Motor Controller and Board PIN connections with their functionalities.

4.5 Ultrasonic Connections

Here is the table depicting relations between the board and ultrasonic sensor:

Motor driver PIN	LPC4088 PIN	Port	Pin Functionality	Reason
GND	PIN1	GND	Ground	Common ground
Trigger	PIN11	P0.9	T2_MAT_3	Send Trigger with Timer2
Echo	PIN16	P0.24	T3_CAP_1	Capture Echo with Timer3
VCC	PIN44	Vout	Power input	5V needed

Table 5: Ultrasonic Sensor and Board PIN connections with their functionalities.

5 Circuit Schematics

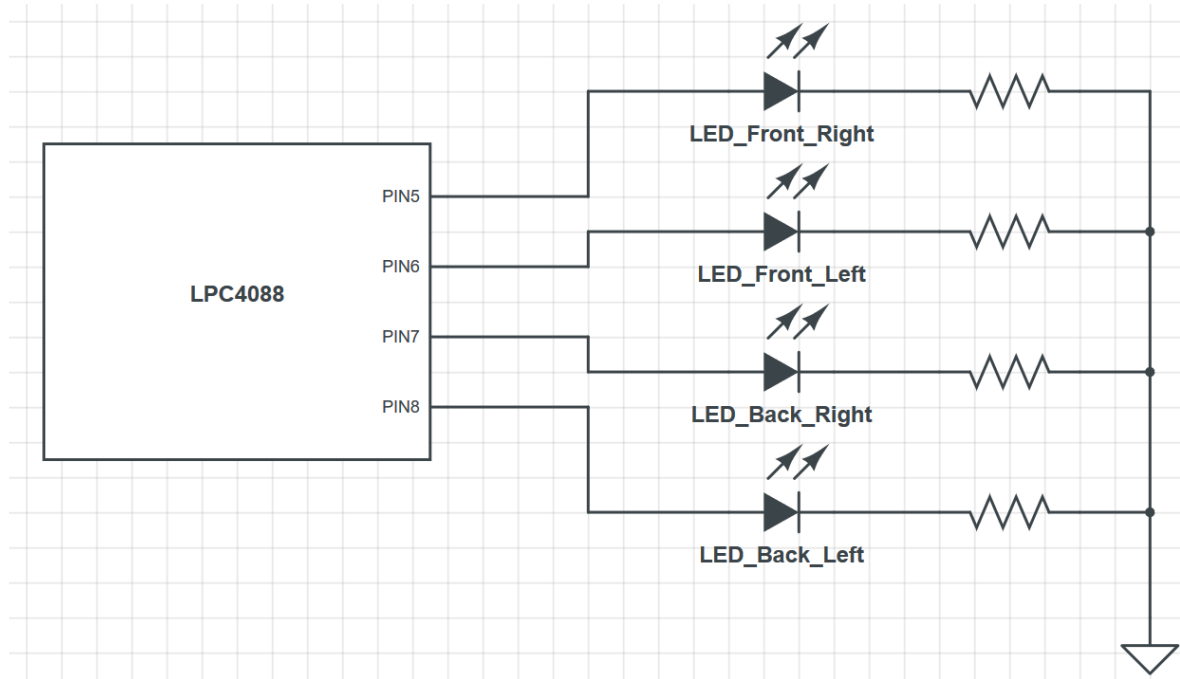## 5.1 LED Circuit Schematics

Figure 5: LED Circuit Diagram.

5.2 LDR Circuit Schematics
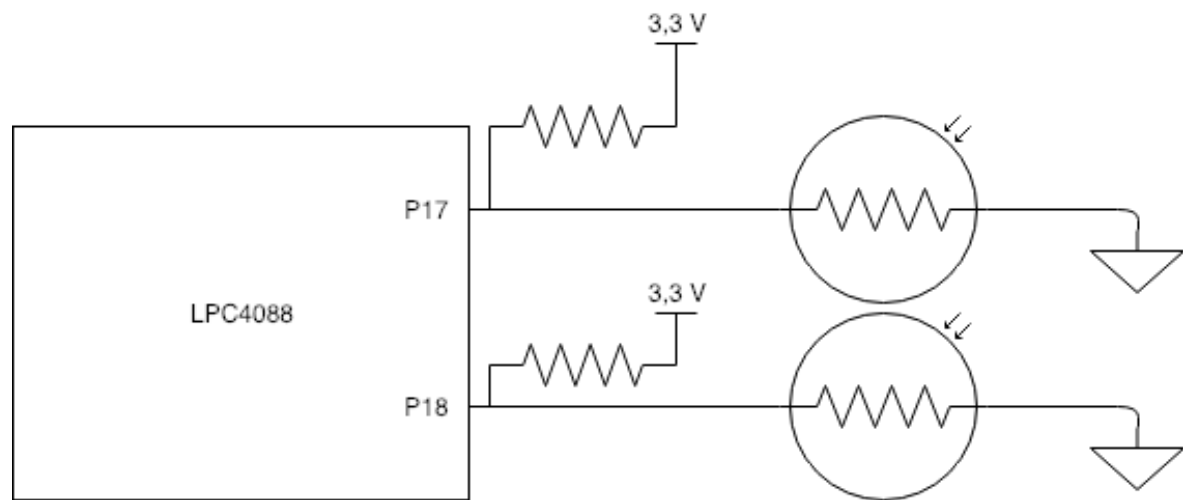

Figure 6: LDR Circuit Diagram.

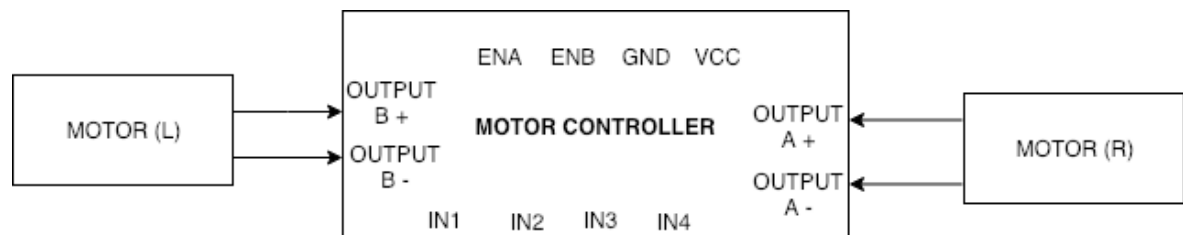## 5.3 Motor - Motor Controller Circuit Schematics

Figure 7: Motor - Motor Controller Circuit Diagram.

5.4   Motor Controller - Board Circuit Schematics
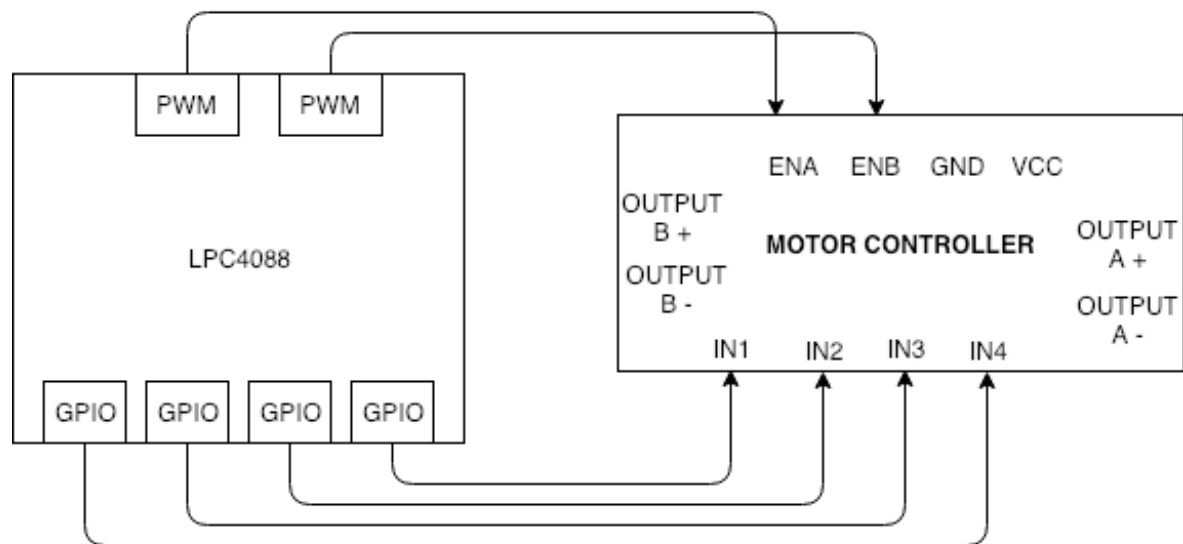
Figure 8: Motor Controller - Board Circuit Diagram.

6 Expense List

There are not much expense.

Name	Expense
Cables	5 TRY
Document Printing	3 TRY

Table 6: Expenses

7 Pseudocodes

Algorithm 1 init

    Initilaze LED
    Initilaze PWM
    Initilaze Joystick
    Initilaze Timer
    Initilaze ADC
    Initilaze UART
    Initilaze UARTInterrupt
    Initilaze MotorController
    Initilaze UltrasonicTriggerTimer
    Initilaze UltrasonicCaptureTimer
    Initilaze ExternalInterrupt

Algorithm 2 Manuel

---

  if Forward Flag **then**
    if ultrasonicSensorDistance < obstacleDistance **then**
      isApproachedTo15 ← true
      Set Backward Flag and Motor and Led
    end if
    if leftLDR < Treshold **then**
      Set Right Motor and Led
    else if rightLDR < Treshold **then**
      Set Left Motor and Led
    else
      Set Forward Motor and Led
    end if
  end if
  if isApproachedTo15 and ultrasonicSensorDistance > escapeDistance **then**
    isApproachedTo15 ← false
    Set Forward Flag and Motor and Led
  end if
  if Joystick Up Pressed **then**
    Set Forward Flag and Motor and Led
  else if Joystick Right Pressed **then**
    Set Right Flag and Motor and Led
  else if Joystick Left Pressed **then**
    Set Left Flag and Motor and Led
  else if Joystick Down Pressed **then**
    Set Backward Flag and Motor and Led
  else if Joystick Center Pressed **then**
    Set Stop Flag and Motor and Led
  end if
  if Right Flag **then**
    if TIMER3→TC mod SECOND < SECOND / 2 **then**
      RightLED
    else
      OffLED
    end if
  end if
  if Left Flag **then**
    if TIMER3→TC mod SECOND < SECOND / 2 **then**
      LeftLED
    else
      OffLED
    end if
  end if

Algorithm 3 Auto

---

    if Joystick Up Pressed or autoDirection > 0 **then**
      autoDirection $\leftarrow$ Forward
      if (leftLDR $>$ actLDR and RightLDR $>$ actLDR) or LDRDataLeft$-$LDRDataRight $<$ toleranceLDR or LDRDataRight$-$LDRDataLeft $<$ toleranceLDR **then**
        Set Forward Flag and Motor and Led
        inflateLeft $\leftarrow 0$
        inflateRight $\rightarrow 0$
      else if LDRDataLeft $<$ actLDR and LDRDataRight $>$ actLDR **then**
        Set Right Flag and Motor and Led
        autoDirection $\leftarrow$ Right
        inflateRight
      else if LDRDataRight $<$ actLDR and LDRDataLeft $>$ actLDR **then**
        Set Left Flag and Motor and Led
        autoDirection $\leftarrow$ Left
        inflateLeft
      end if
      if inflateRight > 500 or inflateLeft > 500 **then**
        $inflatedSpeed \leftarrow TrimpotDataValue/2$
      else
        $inflatedSpeed \leftarrow TrimpotDataValue$
      end if
      RightMotorSpeed $\leftarrow$ inflatedSpeed
      LeftMotorSpeed $\leftarrow$ inflatedSpeed
      if ultrasonicSensorDistance $<$ obstacleDistance **then**
        isApproachedTo15 $\leftarrow$ true
        Set Backward Flag and Motor and Led
      end if
      if isApproachedTo15 and ultrasonicSensorDistance $>$ escapeDistance **then**
        isApproachedTo15 $\leftarrow$ false
        Set Forward Flag and Motor and Led
      end if
      if autoDirection is Right **then**
        if TIMER3$\rightarrow$TC mod SECOND $<$ SECOND / 2 **then**
          RightLED
        else
          OffLED
        end if
      end if
      if autoDirection is Left **then**
        if TIMER3$\rightarrow$TC mod SECOND $<$ SECOND / 2 **then**
          LeftLED
        else
          OffLED
        end if
      end if
    end if

Algorithm 4 update

Require: TrimpotValue, DriveMode and autoDirection from Interrupt, leftLDR, rightLDR

StartUltrasonicTrigger

RightMotorSpeed ← TrimpotValue

LeftMotorSpeed ← TrimpotValue

if DriveMode is Manuel **then**

    Manuel

else

    Auto

end if

Algorithm 5 main

---

  init

while 1 **do**

    update

end while