

CmpE 443 Final Project Design Document

Group Name : Embedded Artists

Members

Güney Dündar

Mehmet Hakan Kurtoğlu

Barın Özmen

Ali Torğutalp

November 20, 2018

Version 1.00

Contents

1	System Level Structural Diagram (Block Diagram)	2
2	Sequence Diagram	3
3	LED Connections	4
4	Motor - Ultrasonic Sensor Connection	5
5	Motor - Driver Connection	6
6	Driver - Board Connection	7
7	Pseudo codes	8
7.1	main.c	8
7.2	esp8266.c	9
7.3	external.c	10
7.4	pwm.c	10
7.5	wait.c	10
7.6	ultrasonic.c	10
7.7	serial.c	11

1 System Level Structural Diagram (Block Diagram)

In this project we have many input sources such as ldr sensors, ultrasonic sensors, trimpod, joystick, obstacles, uart and light. We have 2 mods autonomous and manual. The output is displayed as motor movement and led lights. The block diagram is explained below.

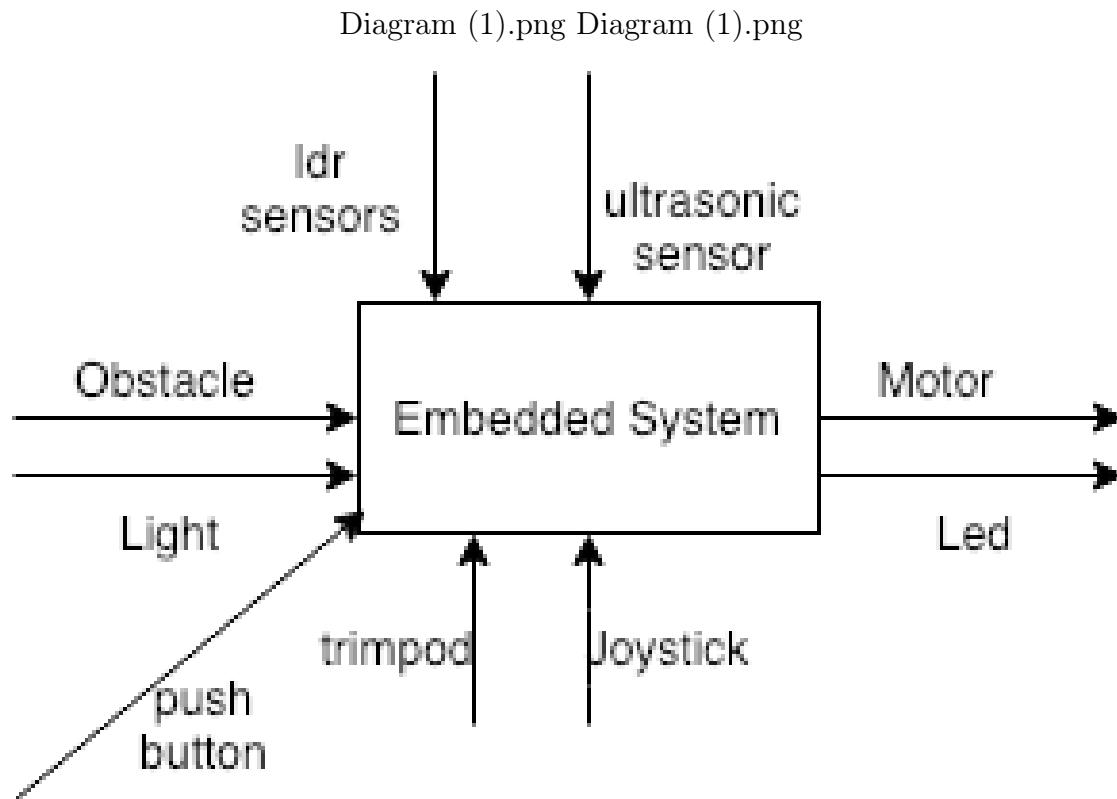
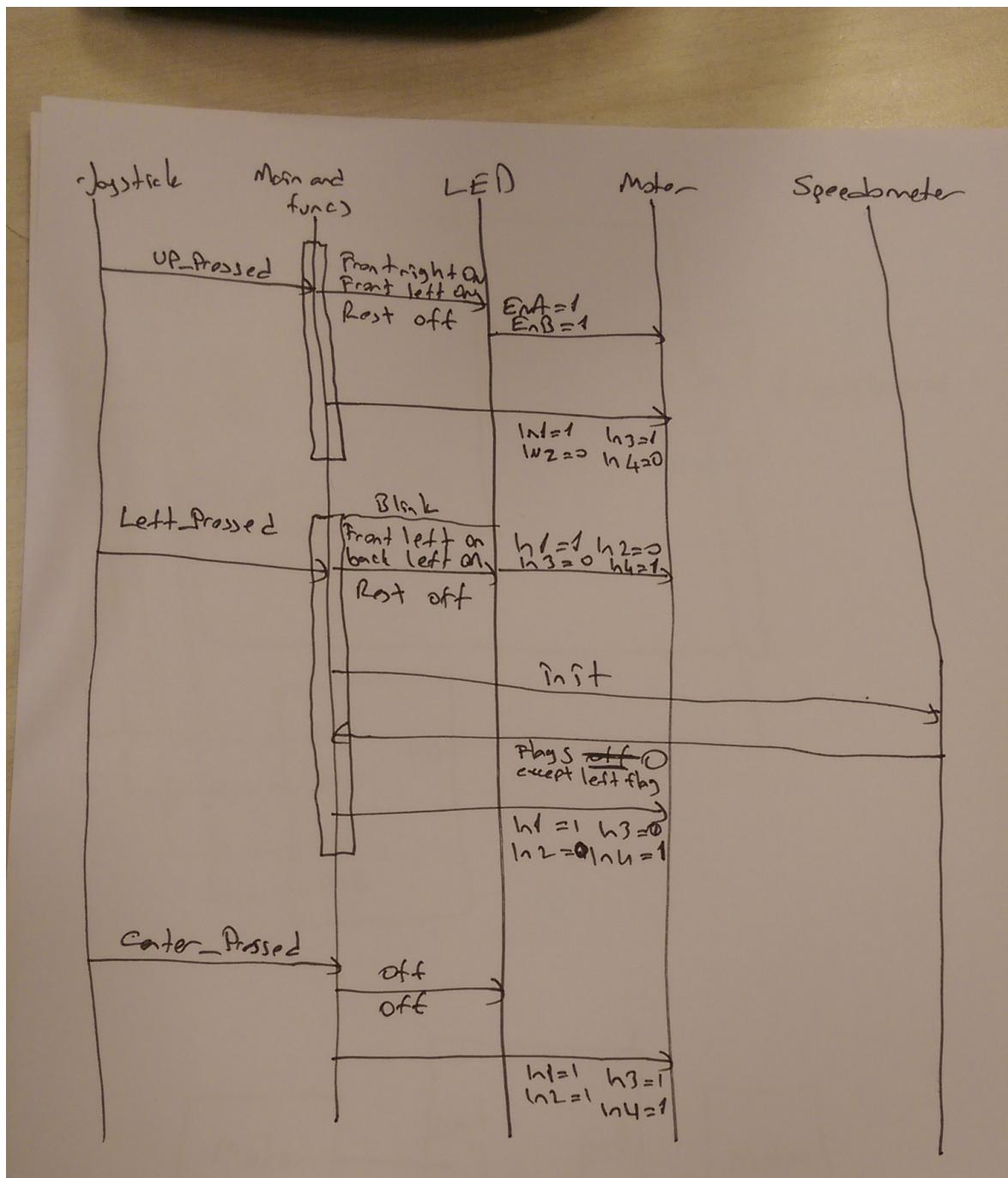


Figure 1: Block diagram.

2 Sequence Diagram



3 LED Connections

We selected P23, P24, P25 and P26 pins as our led pins. Their addresses functionalities and locations on car are shown below. The reason we picked those pins is they are close to each other and can be used as GPIO.

Pin	Address	Location on car	Functionality
P27	P1_6	Front left	GPIO
P24	P1_12	Front right	GPIO
P25	P1_11	Back left	GPIO
P26	P1_7	Back right	GPIO

Table 1: Led pins details

4 Motor - Ultrasonic Sensor Connection

For the motor connection there are 5 pins which are GND, VU, P29, P15, P13 following table shows their addresses connections and functionalities. The reason again was they can be used as GPIO and PWM.

o

Pin	Address	Connection	Functionality
P13	P0_7	IN1	GPIO
P14	P0_8	IN2	GPIO
P11	P0_9	IN3	GPIO
P12	P0_8	IN4	GPIO
P29	P1_3	EnA	PWM
P29	P1_3	EnB	PWM
GND		GND	GPIO
VU		+12V	GPIO

Table 2: Motor pins details

For the ultrasonic sensor connection there are 4 pins which are P16,P11, VU and GND following table shows their addresses connections and functionalities.The reason was they can be used as GPIO.

Pin	Address	Connection	Functionality
P11	P0_29	TRIG	GPIO
P16	P0_24	ECHO	GPIO
VU		VCC	GPIO
GND		GND	GPIO

Table 3: Ultrasonic sensor details

Pin	Address	Connection	Functionality
P17	P0_25	Left	GPIO
P18	P0_26	Right	GPIO

Table 4: LDR pin details

5 Motor - Driver Connection

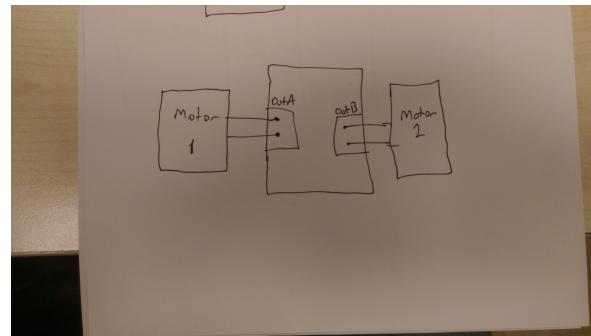


Figure 3: Motor-Driver connections

6 Driver - Board Connection

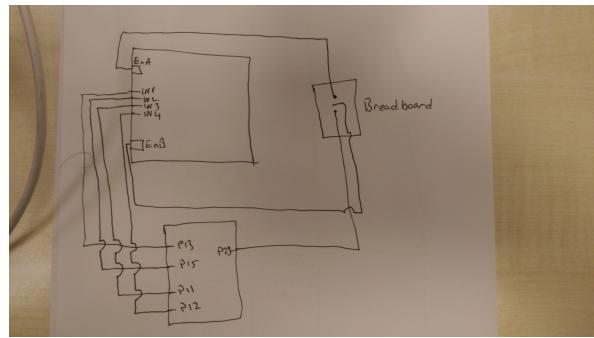


Figure 4: Motor-Driver connections

7 Pseudo codes

7.1 main.c

```

void update
{
    ultrasonicSensorDuration=ultrasonicSensorFallingTime-ultrasonicSensorRisingTime;
    ultrasonicSensorDistance=ultrasonicSensorDuration/58;
    speed=ADCTrimRead();
    PWMWrite(speed * 5);
    LDRleft = ADC_LDRLeftRead();
    LDRright = ADC_LDRRightRead();
    if(printauto)
        serialtransmitdata = auto
        serialwrite(serialtransmitdata)
    elseif(printmanual)
        serialtransmitdata = manuel
        serialwrite(serialtransmitdata)
    if(manualmode)
        if(leftpressed)
            turnleft
        if(rightpressed)
            turnright
        if(uppressed)
            goforward
        if(downpressed)
            gobackward
        if(forwardflag)
            if(LDRleft-LDRright > -threshold & LDRleft-LDRright < threshold)
                goforward
            if(ultrasonicdistance < obstacledistance)
                goback
                while(ultrasonicdistance < obstacledistance      or > threshold)
                    goback
                    ultrasonicSensorDuration = ultrasonicSensorFallingTime-
ultrasonicSensorRisingTime;
                    ultrasonicSensorDistance = ultrasonicSensorDuration/58;
                    goforward
            if(leftldr < threshold)
                turnleft
            if(leftldr > threshold)
                turnright
            elseif(turnleftflag)
                turnleftflag = 1
                leftledson
            elseif(turnrightflag)
}

```

```

turnrightflag = 1
rightledson
elseif(turnbackflag)
    turnbackflag = 1
    backledson
else
    if(joystickUp)
        forwardflag = 1
    if(leftLDR > thresholdrightLDR < threshold)
        forwardFlag = 1
    if(ultrasonicDistance <= obstacleDistance)
        goback
        while(ultrasonicDistance < obstacleDistance)      or      ultraso
200)                                     goback
                                         ultrasonicSensorDuration = ultrasonicSensorFallingTime-
ultrasonicSensorRisingTime;
                                         ultrasonicSensorDistance = ultrasonicSensorDuration/58;
                                         goforward
if(ldrLeft - ldrRight < threshold)
    turnleft
if(ldrLeft - ldrRight > threshold)
    turnright

```

7.2 esp8266.c

```

void esp8266SendCommand
    for(turn until buffer size)
        esp8266ResponsestartIndex = esp8266CurrentBufferIndex;
        esp8266ResponsecurrentIndex = 0;
        ESP8266Write(command);

char esp8266ReadResponse
    while(ESP8266UART->LSR0x01)
        data = ESP8266ReadData()
    if(nonewdata)
        esp8266Buffer[esp8266CurrentBufferIndex] = data
        esp8266CurrentBufferIndex ++
charesp8266ReadData
    while(lsrofuartisnotnull)
        returnnrbrofuart
voidesp8266WriteData
    while(!(ESP8266UART->LSR0x20))
        throfuart = data

```

```

void esp8266Write(char)
    while(thereisdata)
        writethedata
void uart3IRQhandler
    chardata
    data = ESP8266ReadData()
    esp8266Buffer[esp8266CurrentBufferIndex] = data
    esp8266CurrentBufferIndex ++

```

7.3 external.c

```

void eint0IRQhandler
    if(start mode)
        pwmwrite(80)
        forwardFlag=1
    else if(notStartMode)
        startMode=1

```

7.4 pwm.c

```

void pwmCycleRate
    enable pwm match 0 latch
void pwmWrite
    if(value>100)
        value=100
    if(value<100)
        value++
    mr2OFpwm0=tON
    enablePwmMatchRegisterLatch

```

7.5 wait.c

```

void wait(miliseconds)
    for(turn until miliseconds*24000)
void waitMicroSeconds(microseconds)
    for(turn until microseconds*24)

```

7.6 ultrasonic.c

```

void timer2IRQhandler
    if(ultrasonicTriggerEnded=0)
        Change MR3 Register Value for Suggested Waiting

```

```

    Clear pendings for Timer3
    Enable TIMER3IRQn(InterruptRequest)
else
    ultrasonicTriggerEnded = 0
    mr3OfTimer2 = 10
ClearIRRegisterFlagforCorrespondingInterrupt
void timer3IrqHandler
if(ultrasonicSensorEdgeCount = 0)
    ultrasonicSensorEdgeCount = 1
    clearPendingInterrupt
    risingTime = cr1OfTimer3
elseif(ultrasonicSensorEdgeCount = 1)
    fallingTime = cr1OfTimer3
    ultrasonicSensorEdgeCount = 2
    clearPendingInterrupt
    disableIRQ

```

7.7 serial.c

```

char serialReadData
    return RBR of serialUart
void serialWriteData
    thr of serial uart = data
    serialTransmitCompleted=0
void serialWrite(char Data)
    while(there is new data)
        write data
void uart0IrqHandler
if(receive data available interrupt)
    if(*)
        manualMode
    else if()
        autoMode
    else if(66)
        go forward
else if(for thre interrupt)
    if(serialTransmitData>0)
        writeTheData
    else
        serialTransmitCompleted=1

```