

obstacle avoidance robot design v1



kareem magdy albolaqi

sprint

9



Table of Contents

[Project description 2](#_Toc135145406)

[LAYERD ARCHTICTURE 3](#_Toc135145407)

[DRIVERS 5](#_Toc135145408)

[MCAL 5](#_Toc135145409)

[DIO DRIVER 5](#_Toc135145410)

[EXTERNAL INTERRUPT 7](#_Toc135145411)

[TIMER DRIVER 9](#_Toc135145412)

[ICU DRIVER 11](#_Toc135145413)

[HAL 12](#_Toc135145414)

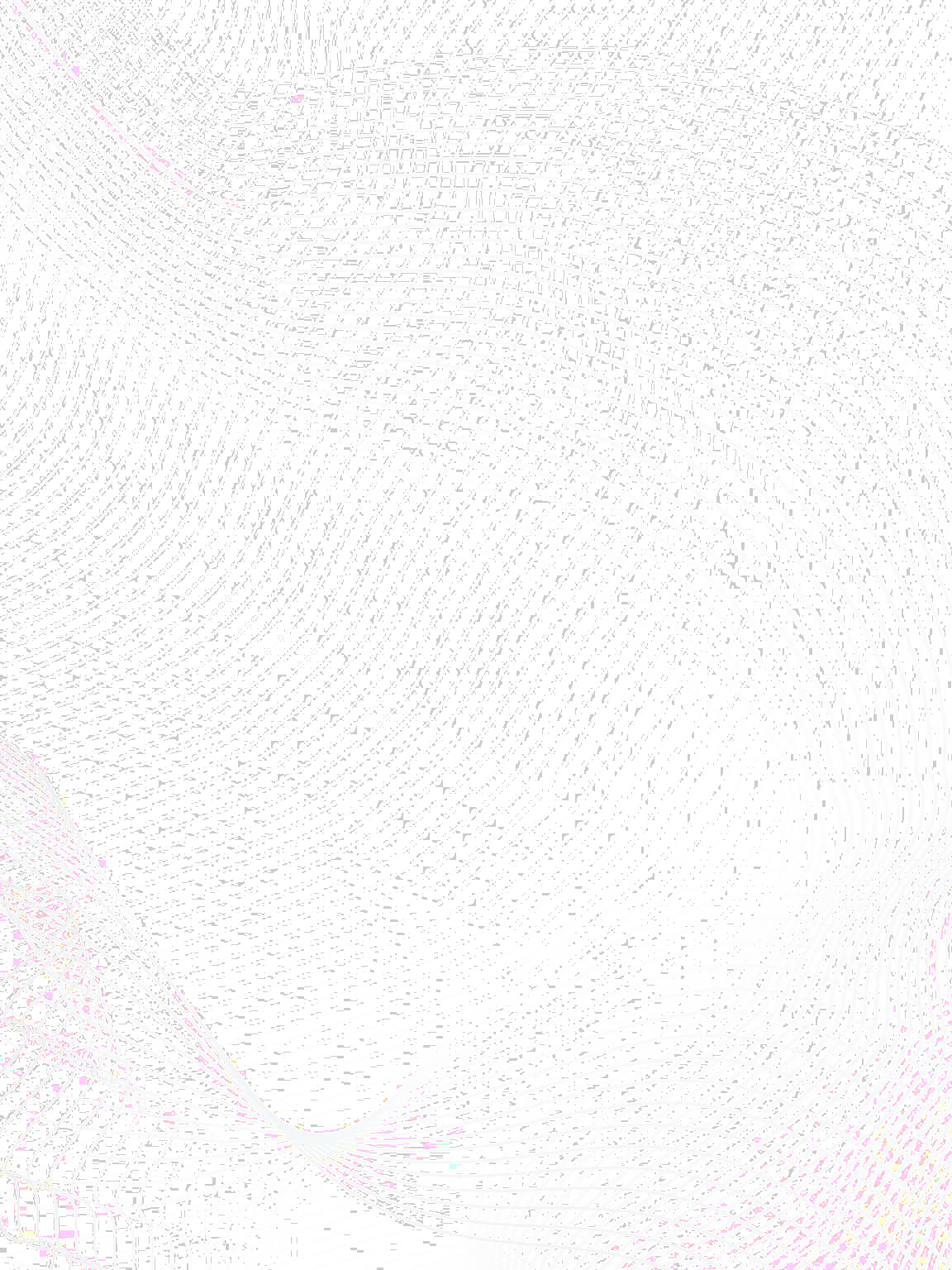
[BUTTON DRIVER 12](#_Toc135145415)

[KEYPAD DRIVER 13](#_Toc135145416)

[LCD DRIVER 15](#_Toc135145417)

[MOTOR DRIVER 18](#_Toc135145418)

[ULTRASONIC DRIVER 19](#_Toc135145419)



* **The car starts initially from 0 speed**
* **The default rotation direction is to the right**
* **Press (Keypad Btn 1), (Keypad Btn 2) to start or stop the robot respectively**
* **After Pressing Start:**

1. The LCD will display a centered message in line 1 “Set Def. Rot.”
2. The LCD will display the selected option in line 2 “Right”
3. The robot will wait for 5 seconds to choose between Right and Left

* **When PBUTTON0 is pressed once, the default rotation will be Left and the LCD line 2 will be updated**
* **When PBUTTON0 is pressed again, the default rotation will be Right and the LCD line 2 will be updated**
* **For each press the default rotation will changed and the LCD line 2 is updated**
* **After the 5 seconds the default value of rotation is set**
* **The robot will move after 2 seconds from setting the default direction of rotation.**
* **For No obstacles or object is far than 70 centimeters:**

1. The robot will move forward with 30% speed for 5 seconds
2. After 5 seconds it will move with 50% speed as long as there was no object or objects are located at more than 70 centimeters distance
3. The LCD will display the speed and moving direction in line 1: “Speed:00% Dir: F/B/R/S”, F: forward, B: Backwards, R: Rotating, and S: Stopped
4. The LCD will display Object distance in line 2 “Dist.: 000 Cm”

* **For Obstacles located between 30 and 70 centimeters**

1. The robot will decrease its speed to 30%
2. LCD data is updated

* **For Obstacles located between 20 and 30 centimeters**

1. The robot will stop and rotates 90 degrees to right/left according to the chosen configuration
2. The LCD data is updated

* **For Obstacles located less than 20 centimeters**

1. The robot will stop, move backwards with 30% speed until distance is greater than 20 and less than 30
2. The LCD data is updated
3. Then preform point 8

**LED SEQUENCE V1.0 PAGE 2**

# Project description

# LAYERD ARCHTICTURE

|  |  |
| --- | --- |
| **UTILITIES** | **APPLICATION** |
| **ECUAL** |
| **MCAL** |
| **MICROCONTROLLER** |

|  |  |
| --- | --- |
| **REGISTERS** | **APPLICATION** |
| **BIT**  **MANIPULATION**  **TYPES** | **ULTRASONIC**  **BUTTON**  **MOTOR**  **KEYPAD** |
| **DIO**  **ICU**  **EXI**  **TIMER** |
| **MICROCONTROLLER** |

# DRIVERS

## MCAL

### DIO DRIVER

Dio\_ErrorStatus Dio\_init(void);

Dio\_ErrorStatus Dio\_WriteChannel(Dio\_Channel channel , Dio\_status state);

Dio\_ErrorStatus Dio\_ReadChannel(Dio\_Channel channel);

|  |  |  |
| --- | --- | --- |
| **DIO\_init** | **DIO\_write** | **DIO\_read** |
|  |  |  |

Configurations :

|  |  |
| --- | --- |
|  |  |

### EXTERNAL INTERRUPT

Ext\_intErrorStatus ExtInt\_init();

Ext\_intErrorStatus INT0\_SetCallback(void(\*callback)(void));

Ext\_intErrorStatus INT1\_SetCallback(void(\*callback)(void));

Ext\_intErrorStatus INT2\_SetCallback(void(\*callback)(void));

|  |
| --- |
| ExtInt\_init(); |
|  |

Configurations :

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  | |
|  | TIMER DRIVER Timer\_ErrorStatus TIMER\_init(Timer\_Mode mode);  Timer\_ErrorStatus TIMER\_start(Timer\_Prescaler prescaler);  void TIMER\_stop(void);  Timer\_ErrorStatus TIMER\_setIntialValue(uint8\_t value);  Timer\_ErrorStatus TIMER\_OvfNum(uint32 overflow); | |
|  | |  |  |  | | --- | --- | --- | | **TIMER\_0\_init** | **TIMER\_0\_start** | **TIMER\_OvfNum** | |  |  |  | | |

Configurations :

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  | |
|  | ICU DRIVER Icu\_ErrorStatus ICU\_init();  Ext\_intErrorStatus INT0\_SetCallback(void(\*callback)(void)); | |
|  | |  |  | | --- | --- | | ICU\_init() | **CALLBACK FINCTION** | |  |  | | |

**Configurations :**

WE CAN USE DIO AND EXI CONFIGURATION TO SET OUR ICU DRIVER CONFIGURATION

## HAL

### BUTTON DRIVER

Dio\_ErrorStatus BUTTON\_read(PORT\_NUM portnum ,PIN\_NUM pinnum, uint8\_t \*value);

|  |
| --- |
| **BUTTON\_read** |
|  |

**Configurations :**

WE CAN USE DIO AND EXI CONFIGURATION TO SET OUR BUTTON DRIVER CONFIGURATION

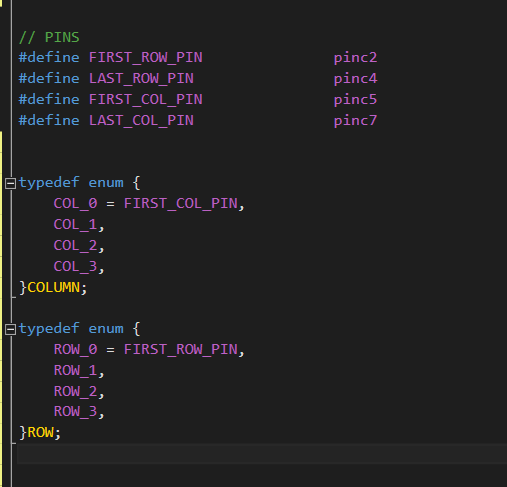
### KEYPAD DRIVER

uint8\_t KEYPAD\_getKey(void);



**Configurations :**

WE CAN USE DIO TO SET OUR KEYPAD DRIVER CONFIGURATION



### LCD DRIVER

void LCD\_WRITE\_COMMAND(uint8\_t a\_COMMAND);

void LCD\_WRITE\_DATA(uint8\_t a\_DATA);

void LCD\_INIT(void);

void LCD\_Clear(void);

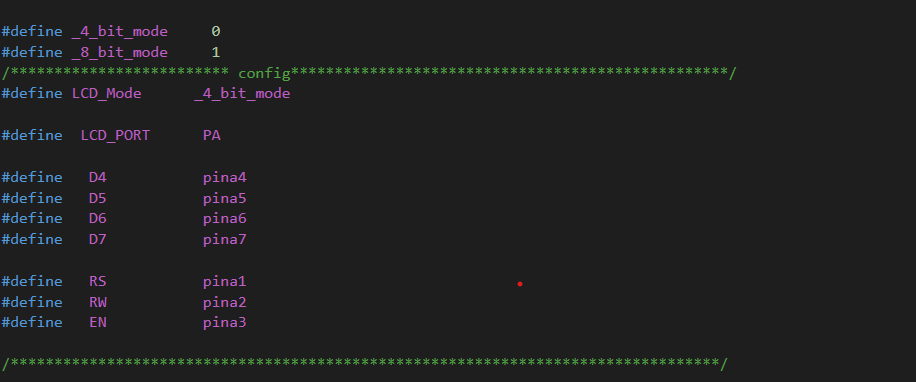
void LCD\_GoTo(uint8\_t a\_line,uint8\_t a\_cell);

void LCD\_Write\_Charecter(uint8\_t a\_char);

|  |  |  |  |
| --- | --- | --- | --- |
| A picture containing text, screenshot, font, number  Description automatically generated |  | A picture containing text, screenshot, font, number  Description automatically generatedA picture containing text, screenshot, font, line  Description automatically generated | |
| A picture containing text, diagram, line, design  Description automatically generated | A picture containing text, screenshot, font, number  Description automatically generated | |  |

**Configurations :**

WE CAN USE DIO TO SET OUR KEYPAD DRIVER CONFIGURATION



### MOTOR DRIVER

void Car\_Moving\_FWD(void);

void Car\_Rotating(void);

void Car\_Stop(void);

|  |  |  |
| --- | --- | --- |
|  |  | A picture containing text, screenshot, font, line  Description automatically generated |

**Configurations :**

WE CAN USE DIO TO SET OUR KEYPAD DRIVER CONFIGURATION

### ULTRASONIC DRIVER

void get\_Distance(uint\_8 \* a\_distance);

