|  |  |
| --- | --- |
| ADocument Title | Specification of DIO Driver |
| **Document Owner** |  |
| **Document Responsibility** | Ahmed Ramadan , Felopater Nagy |
| **Document Identification No** |  |
| **Document Classification** |  |
|  |  |
| **Document Version** | 1.0 |
| **Document Status** |  |
| **Part of Release** |  |
| **Revision** |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Document Change History** | | |
| **Date** | **Version** | **reviewed by** | **Change Description** |
| 16.04.2017 | 1.0 |  | Initial Release |

**Disclaimer**

This specification and the material contained in it is for the purpose of information only. The companies that have contributed to it shall not be liable for any use of the specification.

The material contained in this specification is protected by copyright and other types of Intellectual Property Rights. The commercial exploitation of the material contained in this specification requires a license to such Intellectual Property Rights.

This specification may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only.

For any other purpose, no part of the specification may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

**Table of Contents**

[1 Introduction and functional overview 4](#_Toc511603712)

[1.1 Objective 4](#_Toc511603713)

[1.2 Context Diagram 4](#_Toc511603714)

[1.3 Acronyms and abbreviations 4](#_Toc511603715)

[2 External interfaces 5](#_Toc511603716)

[2.1 Std\_Types.h 5](#_Toc511603717)

[2.1.1 Types 5](#_Toc511603718)

[2.1.2 Interfaces 5](#_Toc511603719)

[2.1.3 Constants 5](#_Toc511603720)

[2.1.4 Variables 5](#_Toc511603721)

[3 Static Design 6](#_Toc511603722)

[3.1 File structure 6](#_Toc511603723)

[3.1.1 Used Files 6](#_Toc511603724)

[3.1.2 File inclusion 6](#_Toc511603725)

[3.2 Types 7](#_Toc511603726)

[3.2.1 Imported types 7](#_Toc511603727)

[3.2.2 Types Definitions 7](#_Toc511603728)

[3.2.2.1 7](#_Toc511603729)

[3.3 Symbol Definition 8](#_Toc511603730)

[3.4 Function definitions 9](#_Toc511603731)

[3.4.1 9](#_Toc511603732)

[3.5 Call-back notifications 10](#_Toc511603733)

[3.6 Scheduled functions 10](#_Toc511603734)

[4 Dynamic Design 11](#_Toc511603735)

[4.1 Mode Management 11](#_Toc511603736)

[4.2 Sequence Diagram 11](#_Toc511603737)

[5 Shared Resources 11](#_Toc511603738)

[6 Configuration specification 11](#_Toc511603739)

[6.1 Containers and configuration parameters 11](#_Toc511603740)

[6.1.1 Variants 12](#_Toc511603741)

[6.1.2 Parameters 12](#_Toc511603742)

[6.1.2.1 12](#_Toc511603743)

[7 Configuration Constraints 12](#_Toc511603744)

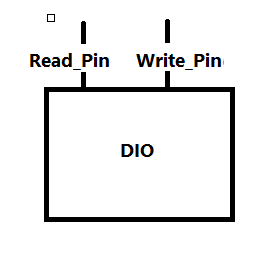
[8 Integration Constraints 13](#_Toc511603745)

# Introduction and functional overview

## Objective

The objective of this module is giving ability to other modules to deal with the pins of microcontroller.

## Context Diagram



## Acronyms and abbreviations

Acronyms and abbreviations that have a local scope.

|  |  |
| --- | --- |
| ***Abbreviation / Acronym:*** | ***Description:*** |
| DIO | Digital input output. |
| Reg | Register. |

# External interfaces

## Std\_Types

### Types

|  |  |
| --- | --- |
| Name | Description |
| U8 | Unsigned char. |
| U16 | Unsigned short Int. |
| U32 | Unsigned long Int. |
| f32 | Float. |
| f64 | Double. |
| S8 | Short char. |
| S16 | Signed short Int. |
| S32 | Signed long Int. |

### Interfaces

### None because DIO in MCAL layer which interact directly with the hardware.

### Constants

None.

### Variables

None.

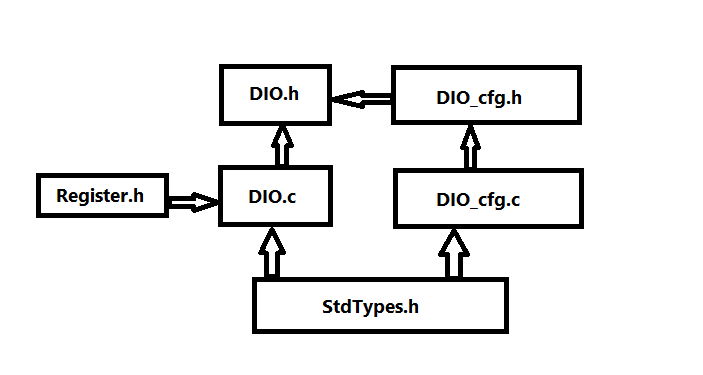
# Static Design

## File structure

### Used Files

|  |  |
| --- | --- |
| ***File*** | ***Description*** |
| DIO.h | This file includes all prototype of function, #defines, extern constant, types which will be accessible to the other modules. |
| DIO.c | This file includes all function implementation and constant. |
| DIO\_cfg.h | This file includes all configurations for DIO module.  EX: output pin, input pin |
| DIO\_cfg.c | This file includes all configuration variable, constant and function which is needed for other files. |

### File inclusion



## Types

### Imported types

|  |  |  |
| --- | --- | --- |
| ***Module*** | ***Imported Type*** | Description |
| DIO | PORT\_NAME | It is enumeration include ports names. |
| MODE | It is enumeration include modes of the pins. |
| LEVEL | It is enumeration include the level of the pin High or Low. |
| DIO\_cfg\_t | It is structure type which save pin mode. |

### Types Definitions

#### 

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name:*** | PORT\_NAME | | |
| ***Type:*** | Enumeration | | |
| ***Range:*** | 0 : 31 |  | From PA0 : PD7 |
| ***Description:*** | It is enumeration include ports names from 0 to 31 and the last one is 32 which referrer to the number of the pins. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name:*** | MODE | | |
| ***Type:*** | Enumeration | | |
| ***Range:*** | 0 : 2 |  | 0->INPUT\_WITH\_PUR  1-> OUTPUT  2> INPUT\_WITHOUT\_PUR |
| ***Description:*** | It is enumeration include modes of the pins input with pull-up resistance ,input without pull-up resistance and output. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name:*** | LEVEl | | |
| ***Type:*** | Enumeration | | |
| ***Range:*** | 0 : 1 |  | 0->LOW  1->HIGH |
| ***Description:*** | It is enumeration include the level of the pin High is one and Low is zero. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name:*** | DIO\_cfg\_t | | |
| ***Type:*** | Structure | | |
| ***Range:*** | 0:31 |  | Every element is consist of mode of the pins. |
| ***Description:*** | It is structure type which save pin mode. | | |

## Symbol Definition

|  |  |
| --- | --- |
| ***Name:*** |  |
| ***Type:*** |  |
| ***Value:*** |  |
| ***Description:*** |  |

## Function definitions

This is a list of functions provided for upper layer modules.

### 

|  |  |  |
| --- | --- | --- |
| ***Service name:*** | DIO\_Init | |
| ***Syntax:*** | void DIO\_Init(void); | |
| ***Service ID[hex]:*** | DIO-001 | |
| ***Sync/Async:*** | Sync. | |
| ***Reentrancy:*** | None. | |
| ***Parameters (in):*** | None. |  |
| ***Parameters***  ***(in/out):*** | None. | |
| ***Parameters (out):*** | None. | |
| ***Return value:*** | None. |  |
| ***Description:*** | This function initiates all register which related to the DIO module.  Ex: DDR, Port, Pin. | |

|  |  |  |
| --- | --- | --- |
| ***Service name:*** | DIO\_WritePin | |
| ***Syntax:*** | DIO\_WritePin(DIO\_PinName name,LEVEL level); | |
| ***Service ID[hex]:*** | DIO-002 | |
| ***Sync/Async:*** | Sync. | |
| ***Reentrancy:*** | None. | |
| ***Parameters (in):*** | name, level. | Name PA0 -> PD7 is the name of the pin.  Level is HIGH or LOW. |
| ***Parameters***  ***(in/out):*** | None. | |
| ***Parameters (out):*** | None, | |
| ***Return value:*** | None. |  |
| ***Description:*** | This function writes the level on the Port register.  Ex: PORTA is HIGH. | |

|  |  |  |
| --- | --- | --- |
| ***Service name:*** | DIO\_ReadPin | |
| ***Syntax:*** | LEVEL DIO\_ReadPin(DIO\_PinName, name); | |
| ***Service ID[hex]:*** | DIO-003 | |
| ***Sync/Async:*** | Sync. | |
| ***Reentrancy:*** | None. | |
| ***Parameters (in):*** | name. | PA0->PD7 is the name of the pin. |
| ***Parameters***  ***(in/out):*** | None. | |
| ***Parameters (out):*** | None. | |
| ***Return value:*** | LEVEL | Level is HIGH or LOW. |
| ***Description:*** | This function returns the input value to the pin through the pin register.  EX:PINA | |

## Call-back notifications

This chapter lists all functions provided by the DIO module to lower layers.

The DIO module does not provide any callback notifications.

## Scheduled functions

This chapter lists all functions called directly by the Basic Software Module Scheduler.

The DIO module has no scheduled functions.

# Dynamic Design

## Mode Management

None.

## Sequence Diagram

## C:\Users\felo\AppData\Local\Temp\DIO sequance Diagram.png

# Shared Resources

There are no shared resources between DIO module and other modules.

There is one global variable between all the function in the module

DIO\_cfg : is an array from type structure which contain the mode of the

Pins of the microcontroller.

# Configuration specification

This chapter defines configuration parameters and their clustering into containers.

## Containers and configuration parameters

The following chapters summarize all configuration parameters.

### Variants

Configuration variants describe sets of configuration parameters:

* VARIANT-PRE-COMPILE (PC)

Only parameters with "Pre-compile time" configuration are allowed in this variant.

* VARIANT-LINK-TIME (LT)

Only parameters with "Pre-compile time" and "Link time" are allowed in this variant.

* VARIANT-POST-BUILD (PB)

Parameters with "Pre-compile time", "Link time" and "Post-build time" are allowed in this variant.

Parameters

### 

### Parameters

#### 

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | DIO\_cfg | | |
| ***Description*** | This parameter is array of type structure which define the mode of the pins of microcontroller. | | |
| ***Multiplicity*** | 1 | | |
| ***Type*** | Scaler | | |
| ***Default value*** | -- OUTPUT | | |
| ***Configuration Class*** | ***Pre-compile time*** | | |
| ***Configuration Class***  ***Scope / Dependency*** | ***Link time*** | X | All Variants |
| ***Post-build time*** | -- |  |
| scope: Global | -- |  |
|  |  | | |

# Configuration Constraints

-The number of pins in the microcon

troller is 32 pin.

-The number of Timer peripheral is three one 16 bit timer and two 8 bit timer.

# Integration Constraints

-Intialization function is required to use the DIO module.