File tag: 01

Digital Input Output Driver Design Document

1. Description:  
   **This is a software driver for Digital Input Output Peripheral of Atmega 32 Microcontroller, this driver was developed by Anas Ebrahim at 25/3/2016 under the supervision of Eng.Mohammad Hassan and Eng.Walid El-Hennawy in the Software Engineering Course.  
   The driver provides the general APIs and Macros needed to use the 32 Digital Input output Pins of the Microcontroller**
2. Driver Architecture:  
   **The driver lies on the MCAL Layer and contains 3 header files  
   1-DIO\_Interface.h which contains the Functions/APIs Prototypes and variable like macros the user can use   
   2-DIO\_Config.h which contains the configuration the user can choose to be the initial directions and values of the input output pins.  
   3-DIO\_private.h which contains macros that is used only inside the driver.  
   And one source file   
   DIO\_Prog.c which contains the Implementation of the driver APIs.**  
   **The Driver also uses (util.h) & (types.h) library from the libraries layer.**
3. Configurations:  
   **The user is required to configure the Initial Direction and Values for each Pin.  
   - The options of the direction should be either (DIO\_u8OUTPUT) or (DIO\_u8INPUT)   
   - The options of the Value should be either (DIO\_u8HIGH) or (DIO\_u8LOW)  
   - The user should choose pins from Range DIO\_u8PIN0 to DIO\_u8PIN31 .**
4. APIs

|  |  |  |  |
| --- | --- | --- | --- |
| APIs | Description | Arguments | Return |
| **DIO\_VoidInit**  **Public** | Initialization function that assigns the Initial Direction and values of the DIO Pins that is configured by the user. | **void** | **void** |
| **DIO\_u8ReadPinVal**  **Public** | Read Input Pin Value function which Takes a Pin index and a pointer to save the pin value at and returns the function state, U8Error if the Index is out of boundary or if the pin is output and u8OK otherwise. | **u8 Copy\_u8PinIdx,  u8\* Copy\_u8PtrToVal** | **U8OK/ u8Error** |
| **DIO\_u8WritePinVal**  **Public** | Write Output Pin Value function which takes a Pin index and the pin value and returns the function state, U8Error if the Index is out of boundary or if the pin is input and u8OK otherwise. | **u8 Copy\_u8PinIdx,  u8 Copy\_u8PinVal** | **U8OK/ u8Error** |
| **DIO\_u8ReadPortVal**  **Public** | Read Input Port Value function which takes a Port index and a pointer to save the port value at and returns the function state, U8Error if the Index is out of boundary or if the port is output and u8OK otherwise. | **u8 Copy\_u8PortIdx,  u8\* Copy\_u8PtrToVal** | **U8OK/ u8Error** |
| **DIO\_u8WritePortVal**  **Public** | Write output Port Value function which takes a Port index and the port value and returns the function state, U8Error if the Index is out of boundary or if the port is input and u8OK otherwise | **u8 Copy\_u8PortIdx,  u8 Copy\_u8PortVal** | **U8OK/ u8Error** |
| **DIO\_u8WritePinDir**  **Public** | Write Pin Direction function which takes a Pin index and the pin direction and returns the function state, U8Error if the Index is out of boundary and u8OK otherwise | **u8 Copy\_u8PinIdx,  u8 Copy\_u8PinDir** | **U8OK/ u8Error** |
| **DIO\_u8WritePortDir**  **Public** | Write Port Direction function which Takes a Port index and the port direction and returns the function state, U8Error if the Index is out of boundary and u8OK otherwise | **u8 Copy\_u8PortIdx,  u8 Copy\_u8PortDir** | **U8OK/ u8Error** |
| **DIO\_u8ReadPinDir**  **Public** | Read Pin Direction function which Takes a Pin index and a pointer to the pin direction and returns the function state, U8Error if the Index is out of boundary and u8OK otherwise | **u8 Copy\_u8PinIdx,  u8\* Copy\_u8PtrToDir** | **U8OK/ u8Error** |
| **DIO\_u8ReadPortDir**  **Public** | Read Port Direction function which Takes a Port index and a pointer to the port direction and returns the function state, U8Error if the Index is out of boundary and u8OK otherwise | **u8 Copy\_u8PortIdx,  u8\* Copy\_u8PtrToDir** | **U8OK/ u8Error** |

1. Shared Variables   
   **There is no shared variables in the driver**
2. Integration constrains  
   **1- The Pin is not reserved for another peripheral that is used in the application   
   2-Choosing the proper configuration of the pin with the external hardware components, the wrong direction of the pin may damage the Microcontroller  
   3-choosing the proper initial value of the output pin that is required to the attached module to the pin**
3. Hardware constrains  
   **1-All the not used pins should be input  
   2-The Pins can drive up to 15mA per pin   
   3- The sum of all pins generated current per port should not exceed 100mA.**
4. APIs Tags

|  |  |
| --- | --- |
| 01\_API1\_S0\_C0\_T0 | **DIO\_VoidInit** |
| 01\_API2\_S0\_C0\_T0 | **DIO\_u8ReadPinVal** |
| 01\_API3\_S0\_C0\_T0 | **DIO\_u8WritePinVal** |
| 01\_API4\_S0\_C0\_T0 | **DIO\_u8ReadPortVal** |
| 01\_API5\_S0\_C0\_T0 | **DIO\_u8WritePortVal** |
| 01\_API6\_S0\_C0\_T0 | **DIO\_u8WritePinDir** |
| 01\_API7\_S0\_C0\_T0 | **DIO\_u8WritePortDir** |
| 01\_API8\_S0\_C0\_T0 | **DIO\_u8ReadPinDir** |
| 01\_API9\_S0\_C0\_T0 | **DIO\_u8ReadPortDir** |
| 01\_API10\_S0\_C0\_T0 | **DIO\_u8ReadPortDir** |