### **Tài liệu hướng dẫn sử dụng DIO Driver theo chuẩn**

### **AUTOSAR Classic**

### **Mục lục**

1. Giới thiệu
2. Cấu trúc dữ liệu và các định nghĩa quan trọng
3. Macro và định nghĩa cần thiết
4. Các hàm API
   * **Dio\_ReadChannel**
   * **Dio\_WriteChannel**
   * **Dio\_ReadPort**
   * **Dio\_WritePort**
   * **Dio\_ReadChannelGroup**
   * **Dio\_WriteChannelGroup**
   * **Dio\_GetVersionInfo**
   * **Dio\_FlipChannel**

## **1. Giới thiệu**

**DIO Driver (Digital Input/Output Driver)** là một phần trong chuẩn **AUTOSAR** (AUTomotive Open System ARchitecture) giúp điều khiển các chân GPIO (General Purpose Input/Output) trên vi điều khiển. DIO Driver hỗ trợ đọc/ghi các chân riêng lẻ, các cổng GPIO và các nhóm chân (channel group).

**Mục đích**: Trừu tượng hóa việc điều khiển phần cứng GPIO, giúp chương trình dễ bảo trì, mở rộng và có khả năng tái sử dụng trên nhiều nền tảng phần cứng khác nhau mà không cần thay đổi mã nguồn chính.

## **2. Cấu trúc dữ liệu và các định nghĩa quan trọng**

#### **Dio\_ChannelType**

* Đại diện cho một channel (một chân GPIO riêng lẻ).
* Kiểu dữ liệu: uint8.

#### **Dio\_PortType**

* Đại diện cho một cổng GPIO (ví dụ: GPIOA, GPIOB).
* Kiểu dữ liệu: uint16.

#### **Dio\_LevelType**

* Đại diện cho mức logic của một channel (mức cao hoặc mức thấp).
* Giá trị:
  + STD\_HIGH: mức logic cao.
  + STD\_LOW: mức logic thấp.
* Kiểu dữ liệu: uint8.

#### **Dio\_PortLevelType**

* Đại diện cho trạng thái logic của toàn bộ cổng GPIO (16-bit).
* Kiểu dữ liệu: uint16.

#### **Dio\_ChannelGroupType**

* Đại diện cho một nhóm các chân GPIO.
* Bao gồm:
  + **port**: ID của cổng (ví dụ: DIO\_PORT\_A).
  + **mask**: Xác định các chân thuộc nhóm.
  + **offset**: Xác định vị trí dịch bit của nhóm.

| typedef struct {  Dio\_PortType port; /\* ID của cổng GPIO \*/  uint16\_t mask; /\* Mask của nhóm kênh \*/  uint8\_t offset; /\* Dịch bit để lấy giá trị chính xác \*/ } Dio\_ChannelGroupType; |
| --- |

## **3. Macro và định nghĩa cần thiết**

#### **Định nghĩa các cổng GPIO**

| #define DIO\_PORT\_A 0 /\* Ánh xạ cho cổng GPIOA \*/ #define DIO\_PORT\_B 1 /\* Ánh xạ cho cổng GPIOB \*/ #define DIO\_PORT\_C 2 /\* Ánh xạ cho cổng GPIOC \*/ #define DIO\_PORT\_D 3 /\* Ánh xạ cho cổng GPIOD \*/ |
| --- |

#### **Macro xác định cổng GPIO dựa trên PortId**

| #define DIO\_GET\_PORT(PortId) \  (((PortId) == DIO\_PORT\_A) ? GPIOA : \  ((PortId) == DIO\_PORT\_B) ? GPIOB : \  ((PortId) == DIO\_PORT\_C) ? GPIOC : \  ((PortId) == DIO\_PORT\_D) ? GPIOD : NULL) |
| --- |

#### **Macro xác định chân GPIO dựa trên ChannelId**

| #define DIO\_GET\_PIN(ChannelId) (1 << ((ChannelId) % 16)) |
| --- |

## **4. Các hàm API**

#### **4.1 Dio\_ReadChannel**

* **Chức năng**: Đọc trạng thái logic của một chân GPIO (DIO Channel).

**Cú pháp**:

| Dio\_LevelType Dio\_ReadChannel(Dio\_ChannelType ChannelId); |
| --- |

* **Tham số**:
  + ChannelId: ID của chân GPIO (DIO Channel) cần đọc. ChannelId có thể được định nghĩa bởi macro DIO\_CHANNEL(GPIOx, Pin) hoặc thông qua các ID cụ thể.
* **Giá trị trả về**:
  + STD\_HIGH: Mức logic cao.
  + STD\_LOW: Mức logic thấp.

**Ví dụ**:

| Dio\_LevelType level = Dio\_ReadChannel(DIO\_CHANNEL\_A0); /\* Đọc trạng thái của GPIOA Pin 0 \*/ |
| --- |

#### **4.2 Dio\_WriteChannel**

* **Chức năng**: Ghi trạng thái logic (cao/thấp) cho một chân GPIO (DIO Channel).

**Cú pháp**:

| void Dio\_WriteChannel(Dio\_ChannelType ChannelId, Dio\_LevelType Level); |
| --- |

* **Tham số**:
  + ChannelId: ID của chân GPIO cần ghi.
  + Level: Trạng thái logic cần ghi (STD\_HIGH hoặc STD\_LOW).

**Ví dụ**:

| Dio\_WriteChannel(DIO\_CHANNEL\_A0, STD\_HIGH); /\* Ghi mức HIGH cho GPIOA Pin 0 \*/ |
| --- |

#### **4.3 Dio\_ReadPort**

* **Chức năng**: Đọc trạng thái logic của toàn bộ cổng GPIO.

**Cú pháp**:

| Dio\_PortLevelType Dio\_ReadPort(Dio\_PortType PortId); |
| --- |

* **Tham số**:
  + PortId: ID của cổng GPIO cần đọc (ví dụ: DIO\_PORT\_A, DIO\_PORT\_B).
* **Giá trị trả về**:
  + Trạng thái logic của toàn bộ cổng GPIO dưới dạng số nguyên 16-bit. Mỗi bit đại diện cho trạng thái logic của một chân GPIO trên cổng.

**Ví dụ**:

| Dio\_PortLevelType portAStatus = Dio\_ReadPort(DIO\_PORT\_A); /\* Đọc trạng thái của toàn bộ GPIOA \*/ |
| --- |

#### **4.4 Dio\_WritePort**

* **Chức năng**: Ghi trạng thái logic cho toàn bộ cổng GPIO.

**Cú pháp**:

| void Dio\_WritePort(Dio\_PortType PortId, Dio\_PortLevelType Level); |
| --- |

* **Tham số**:
  + PortId: ID của cổng cần ghi.
  + Level: Giá trị trạng thái cần ghi (16-bit).

**Ví dụ**:

| Dio\_WritePort(DIO\_PORT\_A, 0xFFFF); /\* Ghi mức HIGH cho toàn bộ GPIOA \*/ |
| --- |

#### **4.5 Dio\_ReadChannelGroup**

* **Chức năng**: Đọc trạng thái logic của một nhóm chân GPIO (DIO Channel Group).

**Cú pháp**:

| Dio\_PortLevelType Dio\_ReadChannelGroup(const Dio\_ChannelGroupType\* GroupIdPtr); |
| --- |

* **Tham số**:
  + GroupIdPtr: Con trỏ đến cấu trúc nhóm kênh DIO (chứa thông tin cổng, mask và offset).
* **Giá trị trả về**:
  + Trạng thái logic của nhóm kênh (16-bit).

**Ví dụ**:

| Dio\_ChannelGroupType myGroup = {DIO\_PORT\_A, 0x00F0, 4}; /\* Nhóm gồm các chân từ A4 đến A7 \*/ Dio\_PortLevelType groupStatus = Dio\_ReadChannelGroup(&myGroup); /\* Đọc trạng thái nhóm \*/ |
| --- |

#### **4.6 Dio\_WriteChannelGroup**

* **Chức năng**: Ghi trạng thái logic cho một nhóm chân GPIO (DIO Channel Group).

**Cú pháp**:

| void Dio\_WriteChannelGroup(const Dio\_ChannelGroupType\* GroupIdPtr, Dio\_PortLevelType Level); |
| --- |

* **Tham số**:
  + GroupIdPtr: Con trỏ đến cấu trúc nhóm kênh DIO.
  + Level: Giá trị trạng thái cần ghi cho nhóm kênh.

**Ví dụ**:

| Dio\_ChannelGroupType myGroup = {DIO\_PORT\_A, 0x00F0, 4}; /\* Nhóm gồm các chân từ A4 đến A7 \*/ Dio\_WriteChannelGroup(&myGroup, 0x0F); /\* Ghi giá trị 0x0F cho nhóm A4-A7 \*/ |
| --- |

#### **4.7 Dio\_GetVersionInfo**

* **Chức năng**: Trả về thông tin phiên bản của DIO driver.

**Cú pháp**:

| void Dio\_GetVersionInfo(Std\_VersionInfoType\* VersionInfo); |
| --- |

* **Tham số**:
  + VersionInfo: Con trỏ đến cấu trúc kiểu Std\_VersionInfoType, chứa thông tin về phiên bản của DIO driver.

**Ví dụ**:

| Std\_VersionInfoType versionInfo; Dio\_GetVersionInfo(&versionInfo); /\* Kiểm tra phiên bản của driver \*/ |
| --- |

#### **4.8 Dio\_FlipChannel**

* **Chức năng**: Đảo trạng thái logic của một chân GPIO (DIO Channel).

**Cú pháp**:

| Dio\_LevelType Dio\_FlipChannel(Dio\_ChannelType ChannelId); |
| --- |

* **Tham số**:
  + ChannelId: ID của chân GPIO cần đảo trạng thái.
* **Giá trị trả về**:
  + Trạng thái logic sau khi đảo (HIGH hoặc LOW).

**Ví dụ**:

| Dio\_LevelType newLevel = Dio\_FlipChannel(DIO\_CHANNEL\_A0); /\* Đảo trạng thái của GPIOA Pin 0 \*/ |
| --- |