### **1. Code API MCAL (PWM Driver)**

Trước tiên, lớp **MCAL** phải có driver **PWM** để điều khiển mô-tơ. Dưới đây là ví dụ về các API của module **PWM** trong MCAL:

#### **File Pwm.h (MCAL)**

| #ifndef PWM\_H #define PWM\_H  #include "Std\_Types.h"  // Cấu trúc cấu hình cho PWM typedef struct {  uint8\_t Pwm\_Channel; // Kênh PWM  uint16\_t Pwm\_Period; // Chu kỳ PWM  uint16\_t Pwm\_DutyCycle; // Tỷ lệ nhiệm vụ (duty cycle) của PWM } Pwm\_ConfigType;  // Prototype cho hàm khởi tạo PWM void Pwm\_Init(const Pwm\_ConfigType\* ConfigPtr);  // Prototype cho hàm cài đặt tỷ lệ nhiệm vụ (duty cycle) void Pwm\_SetDutyCycle(uint8\_t Channel, uint16\_t DutyCycle);  #endif /\* PWM\_H \*/ |
| --- |

#### **File Pwm.c (MCAL)**

| #include "Pwm.h" #include <stdio.h>  // Giả lập khởi tạo PWM void Pwm\_Init(const Pwm\_ConfigType\* ConfigPtr) {  printf("PWM Initialized for Channel %d with Period %d ms and Duty Cycle %d%%\n",   ConfigPtr->Pwm\_Channel, ConfigPtr->Pwm\_Period, ConfigPtr->Pwm\_DutyCycle); }  // Giả lập cài đặt tỷ lệ nhiệm vụ của PWM void Pwm\_SetDutyCycle(uint8\_t Channel, uint16\_t DutyCycle) {  printf("PWM Channel %d set to Duty Cycle: %d%%\n", Channel, DutyCycle); } |
| --- |

### **2. Code Module IoHwAb\_MotorDriver**

#### **File IoHwAb\_MotorDriver.h**

| #ifndef IOHWAB\_MOTORDRIVER\_H #define IOHWAB\_MOTORDRIVER\_H  #include "Std\_Types.h"  // Cấu trúc cấu hình cho mô-tơ typedef struct {  uint8\_t Motor\_Channel; // Kênh PWM điều khiển mô-tơ  uint16\_t Motor\_MaxTorque; // Mô-men xoắn tối đa (Nm) } MotorDriver\_ConfigType;  // Prototype khởi tạo mô-tơ Std\_ReturnType IoHwAb\_MotorDriver\_Init(const MotorDriver\_ConfigType\* ConfigPtr);  // Prototype điều chỉnh mô-men xoắn của mô-tơ Std\_ReturnType IoHwAb\_MotorDriver\_SetTorque(float TorqueValue);  #endif /\* IOHWAB\_MOTORDRIVER\_H \*/ |
| --- |

#### **File IoHwAb\_MotorDriver.c**

| #include "IoHwAb\_MotorDriver.h" #include "Pwm.h" // Gọi API PWM từ MCAL #include <stdio.h> #include <stdlib.h>  // Giả lập cấu hình MotorDriver static MotorDriver\_ConfigType MotorDriver\_CurrentConfig;  // Hàm khởi tạo MotorDriver với cấu hình Std\_ReturnType IoHwAb\_MotorDriver\_Init(const MotorDriver\_ConfigType\* ConfigPtr) {  if (ConfigPtr == NULL) {  printf("Error: Null configuration pointer passed to IoHwAb\_MotorDriver\_Init.\n");  return E\_NOT\_OK;  }   // Lưu cấu hình MotorDriver  MotorDriver\_CurrentConfig.Motor\_Channel = ConfigPtr->Motor\_Channel;  MotorDriver\_CurrentConfig.Motor\_MaxTorque = ConfigPtr->Motor\_MaxTorque;   // Gọi API từ MCAL để khởi tạo PWM  Pwm\_ConfigType pwmConfig = {  .Pwm\_Channel = MotorDriver\_CurrentConfig.Motor\_Channel,  .Pwm\_Period = 1000, // 1 giây (1000ms)  .Pwm\_DutyCycle = 0 // Khởi tạo với duty cycle = 0%  };  Pwm\_Init(&pwmConfig);   // In ra thông tin cấu hình MotorDriver  printf("Motor Driver Initialized with Configuration:\n");  printf(" - Motor Channel: %d\n", MotorDriver\_CurrentConfig.Motor\_Channel);  printf(" - Max Torque: %d Nm\n", MotorDriver\_CurrentConfig.Motor\_MaxTorque);   return E\_OK; }  // Hàm điều chỉnh mô-men xoắn mô-tơ Std\_ReturnType IoHwAb\_MotorDriver\_SetTorque(float TorqueValue) {  // Kiểm tra giá trị mô-men xoắn hợp lệ  if (TorqueValue < 0.0f || TorqueValue > MotorDriver\_CurrentConfig.Motor\_MaxTorque) {  printf("Error: Torque value %.2f Nm out of range (Max: %d Nm).\n", TorqueValue, MotorDriver\_CurrentConfig.Motor\_MaxTorque);  return E\_NOT\_OK;  }   // Tính toán tỷ lệ nhiệm vụ (duty cycle) dựa trên mô-men xoắn  uint16\_t dutyCycle = (uint16\_t)((TorqueValue / MotorDriver\_CurrentConfig.Motor\_MaxTorque) \* 100);   // Gọi API từ MCAL để cài đặt duty cycle của PWM  Pwm\_SetDutyCycle(MotorDriver\_CurrentConfig.Motor\_Channel, dutyCycle);   // In ra giá trị mô-men xoắn đã đặt  printf("Setting Motor Torque to %.2f Nm on Channel %d\n", TorqueValue, MotorDriver\_CurrentConfig.Motor\_Channel);   return E\_OK; } |
| --- |

### **3. Cấu hình và Sử dụng**

#### **File main.c**

| #include "IoHwAb\_MotorDriver.h" #include <stdio.h>  int main(void) {  // Cấu hình cho MotorDriver  MotorDriver\_ConfigType motorConfig = {  .Motor\_Channel = 1, // Kênh điều khiển 1  .Motor\_MaxTorque = 300 // Mô-men xoắn tối đa 300 Nm  };   // Khởi tạo MotorDriver  if (IoHwAb\_MotorDriver\_Init(&motorConfig) != E\_OK) {  printf("Failed to initialize Motor Driver.\n");  return -1;  }   // Đặt mô-men xoắn cho mô-tơ  if (IoHwAb\_MotorDriver\_SetTorque(150.0f) == E\_OK) {  printf("Motor torque set successfully.\n");  } else {  printf("Failed to set motor torque.\n");  }   return 0; } |
| --- |

### **4. Giải thích Luồng Hoạt Động**

1. **Khởi tạo MotorDriver**:
   * Hàm IoHwAb\_MotorDriver\_Init được gọi để khởi tạo mô-tơ với cấu hình định sẵn. Cấu hình bao gồm kênh PWM (được sử dụng để điều khiển mô-tơ) và giá trị mô-men xoắn tối đa.
   * **API Pwm\_Init** từ MCAL được gọi để khởi tạo PWM cho kênh điều khiển tương ứng.
2. **Đặt mô-men xoắn cho mô-tơ**:
   * Khi cần điều chỉnh mô-men xoắn, ứng dụng gọi hàm IoHwAb\_MotorDriver\_SetTorque. Hàm này tính toán tỷ lệ nhiệm vụ (**duty cycle**) dựa trên giá trị mô-men xoắn mong muốn và giá trị mô-men xoắn tối đa.
   * **API Pwm\_SetDutyCycle** từ MCAL được gọi để thiết lập tỷ lệ nhiệm vụ của PWM trên kênh điều khiển, từ đó điều khiển mô-tơ đạt được mô-men xoắn mong muốn.
3. **Kết quả**:
   * Mô-tơ sẽ điều chỉnh tỷ lệ nhiệm vụ của tín hiệu PWM dựa trên mô-men xoắn mong muốn, giúp điều khiển công suất và tốc độ của mô-tơ.

### **5. Kết quả Mô phỏng**

Khi chạy chương trình trên một hệ thống giả lập, đầu ra có thể như sau:

| PWM Initialized for Channel 1 with Period 1000 ms and Duty Cycle 0% Motor Driver Initialized with Configuration:  - Motor Channel: 1  - Max Torque: 300 Nm PWM Channel 1 set to Duty Cycle: 50% Setting Motor Torque to 150.00 Nm on Channel 1 Motor torque set successfully. |
| --- |