

# STM32 | DC Motor Direction Control

---

## Objective

To control the direction (Forward / Reverse) of a DC motor using STM32 Blue Pill and the L293D motor driver, without PWM speed control.

---

## L293D Basics Recap

The L293D is a dual H-Bridge IC.

We'll use one H-Bridge (Motor 1 side).

L293D Pin	Function	STM32 Connection
1 (EN1)	Enable Motor 1	+5V (Always ON)
2 (IN1)	Input 1	PA0
7 (IN2)	Input 2	PA1
3 (OUT1)	Output 1	Motor Terminal 1
6 (OUT2)	Output 2	Motor Terminal 2
8 (Vcc2)	Motor Power (6–12V)	+12V
16 (Vcc1)	Logic Power	+5V
4,5,12,13 (GND)	Ground	Common GND with STM32

---

## Working Principle

- Motor direction is controlled by the logic level of IN1 and IN2.

IN1	IN2	Motor Rotation
1	0	Forward
0	1	Reverse
0	0	Stop (Coast)
1	1	Stop (Brake)

---

## Circuit Diagram

- ◆ Connect STM32 and L293D as follows:

STM32	L293D	Description
PA1	IN1	Motor Direction 1
PA2	IN2	Motor Direction 2
5V	EN1	Motor Enable (Always High)
12V	Vcc2	Motor Supply
GND	GND	Common Ground
OUT1 & OUT2	Motor	Motor terminals

---

## STM32 Code (No PWM, Only GPIO)

```
#include "main.h"

static void MX_GPIO_Init(void);

int main(void)
{
    HAL_Init();
    MX_GPIO_Init();

    while (1)
    {
        // --- Motor Forward ---
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_0, GPIO_PIN_SET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
        HAL_Delay(3000);

        // --- Stop Motor ---
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_0, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
        HAL_Delay(1000);

        // --- Motor Reverse ---
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_0, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_SET);
        HAL_Delay(3000);

        // --- Stop Motor ---
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_0, GPIO_PIN_RESET);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);
```

```

        HAL_Delay(1000);
    }

static void MX_GPIO_Init(void)
{
    __HAL_RCC_GPIOA_CLK_ENABLE();

    GPIO_InitTypeDef GPIO_InitStruct = {0};
    GPIO_InitStruct.Pin = GPIO_PIN_1 | GPIO_PIN_2;
    GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
    GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
    HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
}

```

---

## Explanation

Line	Description
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_0, GPIO_PIN_SET);	Sets IN1 = HIGH
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_1, GPIO_PIN_RESET);	Sets IN2 = LOW → Motor runs forward

---

## Proteus Simulation Steps

1. Open Proteus.
2. Add:
  - o STM32F103C8
  - o L293D
  - o DC Motor
  - o Power supply (12V)
3. Connect as per circuit.
4. Load the .hex file from your CubeIDE build folder.
5. Run simulation →  
The motor will:
  - o Run forward (3 sec)
  - o Stop (1 sec)
  - o Run reverse (3 sec)
  - o Stop (1 sec)
  - o Repeat loop.

## Viva / Interview Questions

Question	Answer
What is the function of EN1?	Enables H-Bridge output (keep HIGH to enable).
Why connect GND of STM32 and L293D?	To maintain a common reference voltage.
How can we stop the motor?	Set both IN1 and IN2 to LOW or HIGH.
What happens if IN1 = IN2?	Motor stops — LOW = coast, HIGH = brake.

---