Experiment-9

Aim: Hands-on experimentation of ATMega32 programming in C.

Objectives: After successfully completion of this experiment students will be able to,

- Use C language for ATMega32 microcontroller programming on AVRStudio.
- Experiment with Timer of ATMega32 on ATMega32 AVR Development Board.

Equipment required:

- Windows7 or later based host computer
- ATMega32 Development board
- USBasp Programmer
- Jumper Wires
- LED

Software required:

- AVR Studio7 installation setup
- USBasp driver installation setup

Theory:

Basics of DC Motor

DC Motor

DC motor converts electrical energy in the form of Direct Current into mechanical energy.

In the case of the motor, the mechanical energy produced is in the form of a rotational movement of the motor shaft.

The direction of rotation of the shaft of the motor can be reversed by reversing the direction of Direct Current through the motor.

The motor can be rotated at a certain speed by applying a fixed voltage to it. If the voltage varies, the speed of the motor varies.

Thus, the DC motor speed can be controlled by applying varying DC voltage; whereas the direction of rotation of the motor can be changed by reversing the direction of current through it.

For reversing the current, we can make use of H-Bridge circuit or motor driver ICs that employ the H-Bridge technique or other any other mechanisms.

WHAT IS L293 Motor driver?

We can't connect a DC Motor directly to a microcontroller due to following reasons.

- A microcontroller can't supply the current required for the working of DC Motor. ATmega32 Microcontroller can source or sink currents up to 40mA but a DC Motor needs current very much more than that.
- The negative voltages created due to the back emf of the motor may affect the proper functioning of the microcontroller.
- You may need to control the direction of rotation of the motor by changing the polarity of the motor supply.
- The operating voltage of the DC Motor may be much higher than the operating voltage of the microcontroller.

To solve these problems you may use transistorized H Bridge in which freewheeling diodes are used to avoid problems due to back emf. Thus it requires minimum four transistors, diodes and resistors for each motor. It is better to use readymade ICs such as L293D or L293 instead of making your own H Bridge, which simplifies your project.

L293D is a Quadruple Half H-Bridge driver commonly used for motor driving. We needn't connect any transistors, resistors or freewheeling diodes. All the four outputs of this IC are TTL compatible and output clamp diodes are provided to drive inductive loads. L293D can provide up to 600mA current, in the voltage raging from 4.5 to 36v. L293 is a similar IC which can provide up to 1A in the same voltage range.

L293 or L293D contains four Half H Bridge drivers and are enabled in pairs. Input EN1 is used to enable pair 1 (IN1-OUT1, IN2-OUT2) and input EN2 is used to enable pair 2 (IN3-OUT3, IN4-OUT4). We can drive two DC Motors with one L293D, but here for demonstration we are using only one. You can connect second DC Motor to driver pair 2 according to your needs.

L293 is a most popular and less expensive built-in H-bridge in a small integrated circuit used for low current motors. H-bridge is a motor driving unit used to control the direction of two motors at a time either clockwise or anticlockwise direction. It is a16 pin IC in which pins out1, out2, out3 and out4 are connected to two motors. Pin EN1 & EN2 are PWM pins while IN1, IN2, IN3 & IN4 are used to provide signals to the motors.

To make the circuit, take Atmega32, L293 IC, crystal, DC motor and make the connections like below.

- Connect the IN1, IN2, IN3 & IN4 pins of L293 with PB0, PB1, PB2 & PB3 pins of Atmega32.
- Pin EN1 & EN2 pins of L293 are connected to 5V
- Connect the VCC with 12V and GND pins of L293 with ground.
- Set the frequency of crystal and Atmega32 to 16MHz.

The below truth table will help you to develop understandings with the basic movements of motors for building a code.

TRUTH TABLE

MOTOR A	MOTOR B	DESCRIPTION
0	0	Both Motors stop
0	1	Motors move in anticlockwise direction
1	0	Motors move in clockwise direction
1	1	Both Motors stop

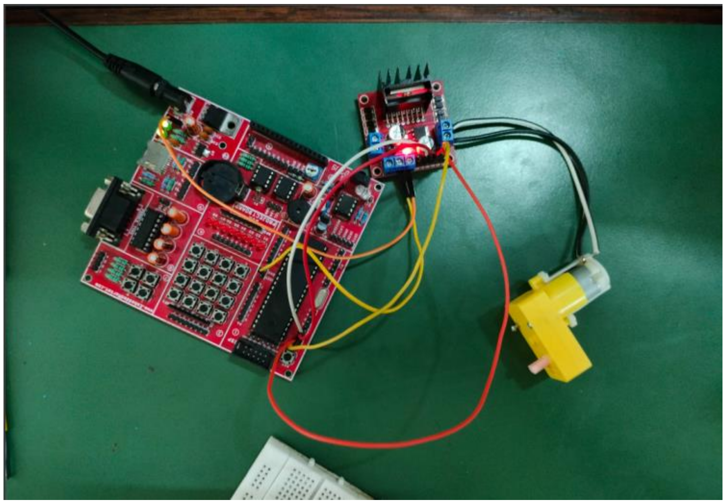
CODE:

```
#ifndef F_CPU
#define F_CPU 16000000UL // 16 MHz clock speed
#endif

#include <avr/io.h>        //standard AVR library
#include <util/delay.h>     // delay library

int main(void) //main starts
{
DDRB= 0xFF; // direction of port B as output
while(1)   //infinite loop
{
PORTB = 0x05; // motor rotation in clockwise direction
_delay_ms(3000);    //delay of 3 sec

PORTB = 0x00;// motor stopped
_delay_ms(1000);    //delay of 1 sec

PORTB = 0x0A; //motor rotation in anticlockwise direction
_delay_ms(3000);    //delay of 3 sec

PORTB = 0x00;// motor stopped
_delay_ms(1000);    //delay of 1 sec

}     //while loop end
}      //main end
```

Description:

As motors are connected to first four pins of PORTB that's why we make the direction of PORTB as output by using command (DDRB). The comments are shown after "//". First motor will rotate in clockwise direction by setting PB.0 & PB.2 high for 3 seconds and then both motors stops for one second by setting all four pins to low. For anticlockwise direction set PB.3 & PB.1 high. The code will execute infinite times due to while loop.

OUTPUT:



CONCLUSION:

By Performing this experiment I came to know about DC Motor interfacing with ATMega32.

Experiment-11 Post Lab Exercise

Student Name: Aryan Langhanoja
Enrollment No: 92200133030

Answer the following questions:

1) What is the need of L293 Driver in this experiment?

- ➢ The L293 driver is needed in this experiment to control the direction and speed of the DC motor. It acts as an H-bridge motor driver, allowing the microcontroller (in this case, AVR ATMega32) to control the polarity of the voltage applied to the motor terminals, thus controlling its direction of rotation.

2) If in the above code, I want to rotate the dc motor clockwise for 5 seconds and anti-clockwise for 2 seconds then what changes will I make in the code?

- ➢ To rotate the DC motor clockwise for 5 seconds and anti-clockwise for 2 seconds in the provided code, you would modify the duration of the motor's rotation in the code. You might also need to adjust the logic for setting the direction of rotation depending on how it's implemented in your code.

3) What is the voltage required for the dc motor to rotate?

- ➢ The voltage required for the DC motor to rotate depends on the motor's specifications. Typically, DC motors operate within a wide range of voltages, commonly from a few volts to several tens of volts. You should refer to the datasheet or specifications of the particular DC motor you are using to determine the appropriate voltage.

4) Write down some applications in which dc motor is used

- ➢ Some applications in which DC motors are commonly used include:
  - o Robotics (for movement mechanisms)
  - o Electric vehicles (for propulsion)
  - o Industrial automation (for conveyor belts, lifts, etc.)
  - o Home appliances (blenders, fans, etc.)
  - o Aerospace (for various actuation purposes)

5) If I want to rotate the motor clockwise and I have connected IN1, IN2, IN3, and IN4 to PB0, PB1, PB2, PB3 then which values in PORTB will be changed?

- ➢ If you have connected IN1, IN2, IN3, and IN4 to PB0, PB1, PB2, PB3 respectively, and you want to rotate the motor clockwise, you would set the values in PORTB accordingly to activate the appropriate inputs for the L293 driver. Assuming a simple configuration where you're using the pins directly to control the L293 driver, setting PB0, PB2 to HIGH and PB1, PB3 to LOW would rotate the motor clockwise. So, you would set PORTB = 0b00001010.