International Journal of Scientific Engineering and Technology
Volume No.5 Issue No.1, pp: 85-88
ISSN:2277-1581
01 Jan.2016

Design and Implementation of a Digital Tachometer

Md. Masud Rana, Md. Sahabuddin, Shourov Mondol
Department of Electrical and Electronic Engineering, Jessore University of Science and Technology, Jessore-7408, Bangladesh.
Corresponding Email : mmrana.3012@gmail.com

Abstract: ***Tachometer is a measuring instrument used for measuring the speed of a rotating body. The unit of measured speed by tachometer is expressed in revolution per minute or RPM. Tachometers were purely mechanical in past. In that time, the speed measuring parameters were sent to the tachometer through mechanical coupling (cable or shaft) and the rpm is measured using a gear mechanism and it is identified using needle mechanism. The tachometers have changed a lot due to rapid development of modern electronics. This article is for a contactless digital tachometer using Arduino. Here the speed of the motor is also controlled. The RPM, the duty cycle and other information are displayed on a LCD screen.***

Keywords— Tachometer, LCD-Display, RPM, Arduino, IR LED, Phototransistor, Diode, Light, Speed.

I. Introduction

Tachometer is an essential instrument in this modern era of industrialization. In industry it is necessary to control the speed of motor at a definite RPM and for this counting of RPM is essential. Tachometer is an instrument used for measuring the number of revolution of an object in a given interval of time. Usually it is expressed in revolution per minute or RPM. Earlier tachometers are purely mechanical where the revolution is transferred to the tachometer through mechanical coupling (cable or shaft), the rpm is determined using a gear mechanism and it is displayed on a dial. With the advent of modern electronics, the tachometers have changed a lot. This project is about a contactless digital tachometer using Arduino. The speed of the motor can also be controlled with the circuit.

Project Objective:

The project aims at designing a system to count the speed of a motor in revolution per minute and to control the speed of a dc motor. In industry for various purposes it is necessary to control the speed of motor at a specific revolution per minute (RPM).

Block Diagram:

The block diagram of this digital tachometer includes the sensor part, control circuit, Arduino-UNO board, a LCD display and a dc motor.

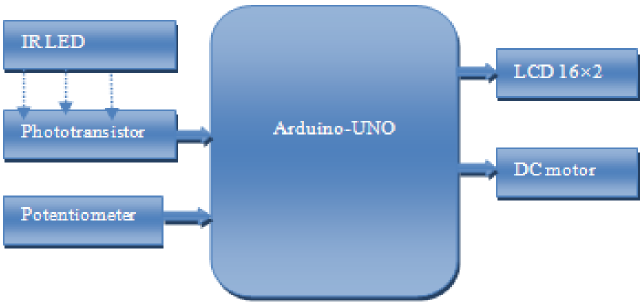

Fig. 1: Block diagram of a digital tachometer.

II. Hardware and Software Tools

Hardware Tools:

- Arduino-UNO board.
- IR (infrared) transmitter diode.
- Phototransistor.
- LCD display.
- Transistor (2N2222).
- Resistors (100Ω, 22kΩ).
- Potentiometer (10kΩ).
- Diode (1N4007).

Software Tools:

- Arduino 1.0.5 r2.

A Brief History of Arduino-UNO Board:

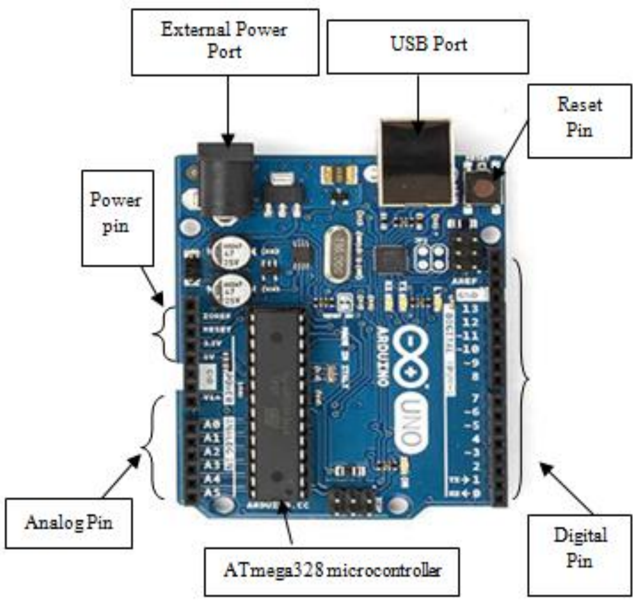

Fig. 2: Arduino-UNO board [1].

Arduino-UNO at a glance:

The Arduino Uno is a microcontroller based board. It contains the ATmega328 microcontroller. This board contains 14 digital input/output pins. Among these pins, 6 pins can be used as PWM outputs. It has 6 analog inputs, a USB connection, a power jack, an ICSP header, and a reset button. It has 16 MHz of clock speed. It can simply be connected to a computer with a USB cable or with a AC-to-DC adapter or battery to get started. The main functioning component of this board is the ATmega328 microcontroller. The ATmega328 microcontroller has 2KB of SRAM, 1KB of EPROM and 32KB of flash memory [2], [3].

Power connection to Arduino-UNO:

The Arduino Uno can be powered via the USB connection or with an external power supply. Leads from a battery can be inserted in the Ground and Vin pin headers of the POWER connector. The board can operate on an external supply of 6 to 20V. If supplied with less than 7V, however the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts [1].

Pin mode definition:

There are 14 digital pins on the board. Each of the pin can be used as input or output using pinMode(), digitalWrite() [5], and digitalRead() [7] functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kΩ. Using 0 (RX) and 1 (TX) pins serial TTL data can be received (RX) and transmitted (TX). External interrupt pins are pin 2 and 3. These pins can be used to trigger an interrupt on a low value, a rising or falling edge, or a change in value using the attachInterrupt() [6] function. The pin number 3, 5, 6, 9, 10 and 11 provide 8-bit PWM output with the analogWrite() [4] function. The 6 analog input pins A_0 through A_5 provide 10 bits resolution that means 1024 different values.

III. RPM Counting

For the purpose of RPM counting we use the property of phototransistor that when IR light falls on the phototransistor its output becomes high and low when interruption occurs. The microprocessor counts the number of interrupts and time in millisecond. Ultimately we can measure the number of revolution per unit time by using this formulae --

NR = (number of interrupts/time in milliseconds)………..(i)

Where NR = number of revolution.

We can count RPM by converting the time interval in equation (i) in minute as the following

RPM = (number of interrupts/time in milliseconds) × 20000…….(ii)

IV. Circuit Description and Connection diagram

For sensing the RPM we use the special alignment of IR LED and IR phototransistor. The IR LED and IR phototransistor are aligned in such a way that light from the LED falls on the transistor. This combination of IR LED and IR phototransistor forms the sensor part of this project. To get output from the transistor the base of the transistor is left free. The emitter of the transistor is grounded through a 22 KΩ resistor. The output voltage across the resistor is taken as input to Arduino-UNO through interruption pin 3. We have also controlled a dc motor using pulse width modulation and this is performed using PWM output pin 9. For controlling the motor the analog input is controlled by using the A1 pin. For showing the RPM and the duty cycle of PWM we have interfaced a 16×2 LCD display. For LCD interfacing [8] we have used the digital pin 2, 4, 5, 6,11 and 12 using the function LiquidCrystal lcd(12,11,6,5,4,2).

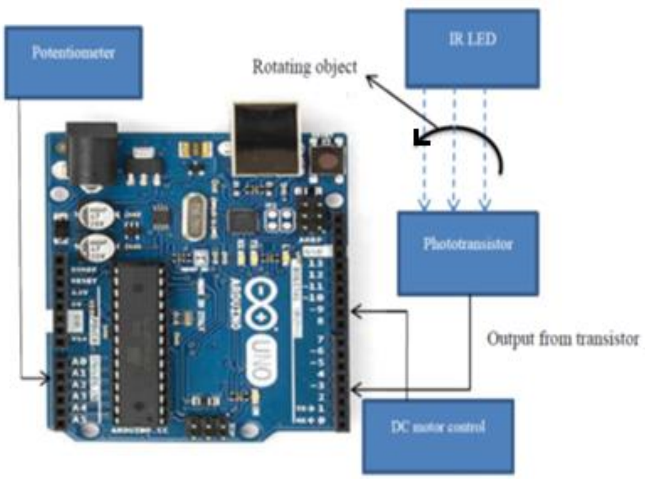

Fig. 3: Connection diagram of a digital tachometer.

V. Flow chart of a Digital Tachometer

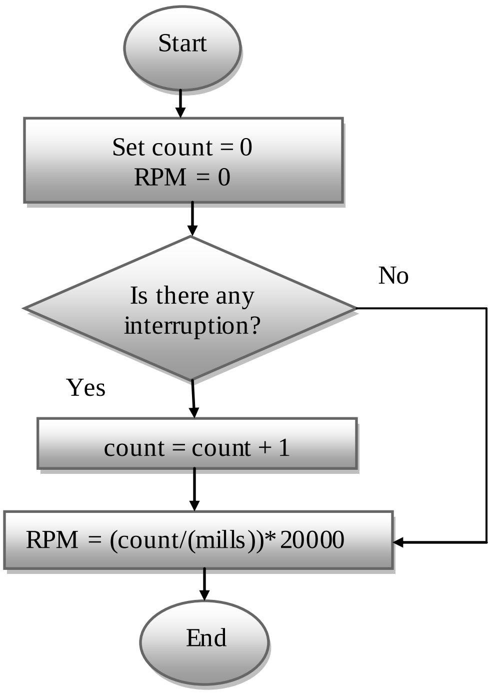

Fig. 4: Flow chart of a digital tachometer

ISSN:2277-1581
01 Jan.2016

VI. Circuit Operation

An IR photo transistor and IR LED forms the sensor .The use of IR phototransistor avoids other light interferences from the environment. The photo transistor and IR diode are aligned side by side. The object of which the speed to be measured is placed between the IR LED and IR photo transistor. Here we use a 9V dc motor for measuring the speed. The phototransistor conducts only when the light falls on it. At the time of rotating the blade of the motor interrupts the light passing from IR LED to IR phototransistor and this interruption is sensed by Arduino. Using the interrupt service routine the number of interrupt is counted. The mills function measures the time for each interruption. Then by using the following formulae the RPM of the motor can be measured.

RPM = (number of interrupts/time in milliseconds) × 20000.

VII. Comparison between existing tachometer in our lab and our implemented one

A tachometer is available in our machine laboratory but problem is that this one does not measure the exact RPM. That is why we intend to design a digital tachometer with less error.

Table 1: Comparison between two tachometers

Duty Cycle	RPM measured with existing tachometer	RPM measured with implemented tachometer
13%	490	513
34%	1550	1599
92%	2795	2820
100%	2930	2968

The motor we used is rated 3000 rpm. So the percentage error of our implemented tachometer is 1.07% and the percentage error of the existing tachometer is 2.33%.

VIII. Some snaps of the implemented project

Fig. 5: Implemented figure of a digital tachometer (Speed = 2820 RPM, PWM=92%)

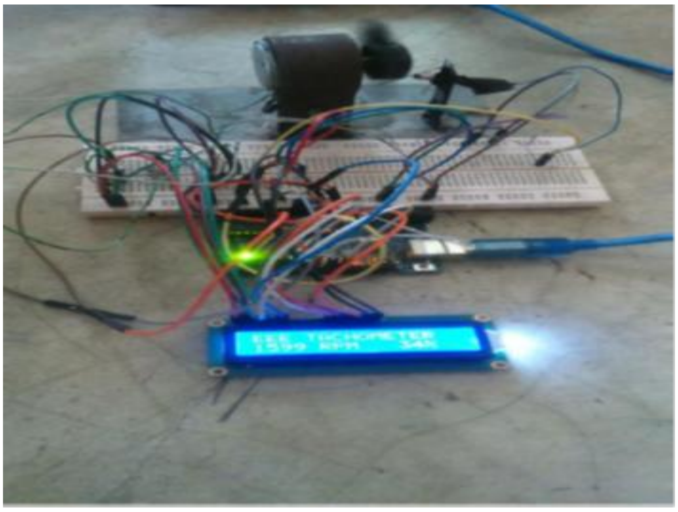

Fig. 6: Implemented figure of a digital tachometer (Speed = 1599 RPM, PWM=34%)

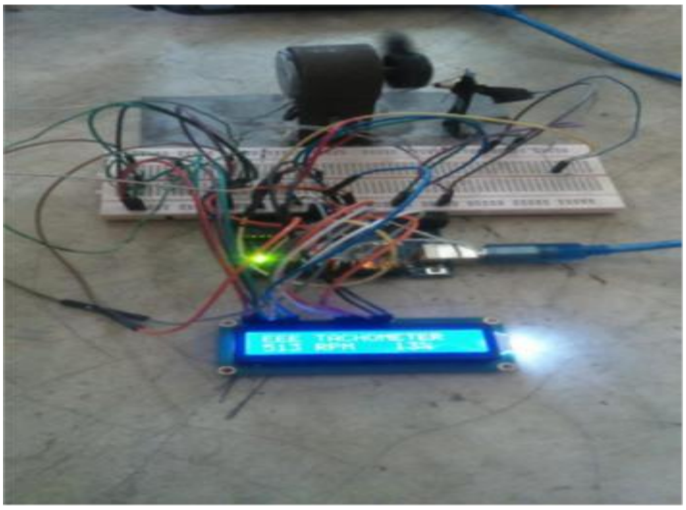

Fig. 7: Implemented figure of a digital tachometer (Speed = 513 RPM, PWM=13%)

IX. Application

Tachometer is an essential instrument in this modern era of industrialization. In industry it is necessary to control the speed of motor at a definite RPM and for this counting of RPM is essential. Tachometer performs this task of counting. Using the theory behind tachometer we can develop other device such as visitor counter, object counter.

X. Conclusion

In this modern era of industrialization motor and generator become part and parcel in industry. For specific application in industry the speed is needed to be fixed and for this reason it is necessary to know the speed in RPM (revolution per minute). Tachometer is such a device to count the RPM (revolution per

ISSN:2277-1581
01 Jan.2016

minute). In our project (Digital Tachometer) we have tried our best to implement digital tachometer in such a way that the count should be accurate.

References

i. *http://www.arduino.cc.(Access time 29/01/2015 . 9:06 PM).*
ii. *https://en.wikipedia.org/wiki/Arduino*
iii. *Arduino cookbook.*
iv. *arduino-1.0.5-r2/reference/AnalogWrite.html*
v. *arduino-1.0.5-r2/reference/DigitalWrite.html*
vi. *arduino-1.0.5-r2/reference/AttachInterrupt.html*
vii. *arduino-1.0.5-r2/reference/DigitalRead.html*
viii. *INTERFACING LCD TO ARDUINO. JHD162A IS THE 16X2 LCD MODULE USED HERE. IT RUNS ON THE HITACHI 44780 DRIVER.HTM*