Contactless Digital Tachometer using IR Sensor

A contactless digital tachometer: an IR reflective sensor counts shaft revolutions and an Arduino computes RPM on a 16x2 LCD with hold and average modes. Delivered built-to-order with sensor head, demo motor, firmware, report, PPT and viva kit.

1. Introduction / Background

Measuring rotational speed without touching the shaft matters wherever contact would disturb the measurement or wear the sensor — motors, turbines, spindles. The industrial answer is non-contact sensing: a reflective mark on the shaft, an optical sensor watching it, and a counter converting pulses per second into RPM.

This project implements exactly that with student hardware: an IR reflective sensor module faces a motor shaft carrying a contrasting mark, each pass generates a pulse, and the Arduino counts pulses over a timed gate — or measures the pulse period — to compute revolutions per minute on a 16x2 LCD.

2. Problem Statement

Students typically estimate motor speed by eye or by timing a few revolutions with a stopwatch — no instrument discipline, no sense of resolution versus gate time. This project builds a complete measurement-instrument chain instead: reflective IR sensing, interrupt-driven counting, unit conversion, hold and averaging modes, and a buyer-run calibration procedure against a reference.

3. Objectives

- Detect shaft revolutions with a reflective IR sensor and a contrasting mark, with no shaft contact or loading.
- Count pulses on a hardware external interrupt for accurate capture even at high RPM.
- Compute and display RPM live on a 16x2 LCD with hold and multi-gate average modes.
- Support 1–4 marks per revolution, firmware-selectable, with a documented calibration procedure.

4. Proposed Methodology

The system is developed in three stages. (1) Sensing: a white mark is fixed on the motor shaft; the IR module is positioned a few millimetres away, and each revolution produces one clean digital pulse via the module's comparator. (2) Counting: the pulse feeds an Arduino external-interrupt pin that increments a counter on every rising edge; once per second the firmware reads the counter, computes RPM as pulses-per-second x 60 divided by marks-per-revolution, and resets. (3) Display and modes: the result is written to the LCD, averaged over several gates in average mode, and frozen by the hold button.

5. System Architecture / Working

- Sensing: reflective IR LED + photodiode with comparator; a few-mm standoff; contrasting shaft mark.
- Counting: external interrupt on rising edge; 1 s gate time; RPM = (pulses/s x 60) / marks-per-revolution.
- Display: 16x2 LCD with live RPM; hold button freezes the reading; mode button cycles marks-per-revolution.
- Calibration: buyer-run procedure comparing against a reference to verify the design-target accuracy.

6. Hardware / Software Requirements

- Arduino Uno (ATmega328P), IR reflective sensor module with comparator, 16x2 character LCD.
- DC demo motor with marked shaft disc, push buttons (hold, mode), 5–9 V power supply.
- Arduino IDE (C/C++ firmware); mounting stand for the sensor head.

7. Expected Outcomes

- A working non-contact tachometer with live RPM readout over the design-target 30–9,999 RPM range.
- Hold and average modes demonstrated on the demo motor.
- A buyer-performed calibration check documenting the design-target accuracy of approximately ±1%.
- A complete report, PPT and viva Q&A; covering optical sensing, interrupts and RPM computation.

8. Applications

- Teaching sensor interfacing, interrupt-driven counting and unit conversion in instrumentation labs.
- Measuring demo-motor speeds in project exhibitions.
- A base platform for frequency-counter or direction-sensing extensions.

9. Future Scope

- Direction sensing with a quadrature pair of sensors.
- SD-card RPM logging with timestamps.
- Wireless display of readings on a phone.
- Extended low-RPM mode with adaptive gate timing.

10. Conclusion

The project delivers a complete measurement instrument — sensing, interrupt counting, conversion and calibration — with its accuracy stated honestly as a design target of approximately ±1% that the buyer verifies with the documented procedure, not as a claimed measured result.
