SUDAN UNIVERSITY OF SCIENCE AND TECHNOLOGY
COLLEGE OF GRADUATE STUDIES

A Design of Downstream Gate Controller for the Pump Turbine in Khashm-Algirba Dam Power Station

تصميم متحكم بوابة أسفل النهر للمضخة التوربينية في محطة خزان خشم القربة

A Thesis Submitted in Partial Fulfilment of the Degree of M.Sc. in Mechatronics Engineering

Prepared by:

Mohammed Abdalgadir Jumaa Mohammed

Supervisor:

Dr. Ali Mohammed Hamdan Adam

March 2018

DEDICATION

To my Family,
To my Friends,
To my Colleagues.

ACKNOWLEDGEMENT

Depth of gratitude and gratefulness is owed to those who have contributed to this work. Their effort will definitely prepare me for my professional life and improve my capacity to contribute to the profession; I assure them at least I will try my best.

I thank Eng. Ali Bashir Alzaky for helping me in this thesis.

I wish to express my gratitude to my supervisor Dr. Ali M. Hamdan Adam for his constant support and encouragement to complete this project.

I also thank Eng. Johr Khalid for assistance me in developing the Model.

I would like also to thank my brothers Abdulazim and Bashir for their continuous support during my course of study and through this study project.

I would have never succeeded in completing this master without cooperation, encouragement and help provided from my wife.

ABSTRACT

In this research a study on the pump turbine was conducted and found that in case the hydraulic system failure to close the wicket gate and runner blade of the pump turbine the speed of the pump turbine will increase and leading to a complete damage for the turbine. The company is forced to replace the turbine and this leads to high cost. A controller of the downstream roller gate for the pump turbine in Khashm-Algirba dam power station was designed and the prototype of the control system was conducted and tested in electronic laboratory kits. According to the results obtained it was found when the speed of the pump turbine reached to 1080 rpm the controller sent a signal for issued sound and light alarm and when the speed reached to 1260 rpm the controller sent signal to the downstream roller gate servo-valve to open it for close the downstream roller gate to stop the pump turbine. The Downstream gate of the pump turbine was closed in 60 seconds, leading to the unit being safely shut down.

مستخلص

في هذا البحث أجريت دراسة عن المضخة التوربينية ووجدت أنه في حالة فشل النظام الهيدروليكي لإغلاق أبواب وريشة الطلمبة التوربينية فإن سرعة التوربينة سوف تزداد وتؤدي إلى تلف كامل للتوربينة وتضطر الشركة لإستبدال التوربين وهذا يؤدي إلى تكاليف عالية. تم تصميم جهاز تحكم للبوابة الخلفية للمضخة التوربينية في محطة توليد كهرباء خزان خشم القربة وكذلك تم عمل نموذج مبدئي لنظام التحكم في الباب الخلفي وإختباره في معمل معدات إلكترونية. وفقا للنتائج التي تم الحصول عليها تبين أنه عندما وصلت سرعة المضخة التوربينية 1080 دورة في الدقيقة أرسلت وحدة التحكم إشارة صوتية وضوئية للتنبيه وعندما وصلت السرعة 1260 دورة في الدقيقة أرسلت وحدة التحكم إشارة لصمام الباب الخلفي للمضخة التوربينية لإغلاق الباب وإيقاف التوربينة. ووجد ان الباب الخلفي للتوربينة يغلق في 60 ثانية مما يؤدي إلي إيقاف الوحدة بأمان.

TABLE of CONTENTS

LIST of TABLES

LIST of FIGURES

LIST of ABBREVIATIONS

DSG	Downstream Gate
PT2	Pump Turbine No. Two
WG	Wicket Gate
CB	Circuit Breaker
DC	Direct Current
IR	Infrared
LCD	Liquid Crystal Display
LED	Light Emitting Diode
MCU	Microcontroller
QFP	Quad Flat Package
DIP	Dual In Line
I/O	Input /Output
ADC	Analog to Digital Converter
AC	Alternating Current
CW	Clock Wise
CCW	Counter Clock Wise
CMOS	Complementary Metal Oxide Semiconductor
EEPROM	Erasable Programmable Read Only Memory
SRAM	Static Random Access Memory
SPI	Serial Peripheral Interface
TWI	Two- Wire Interface
USART	Universal Asynchronous Receiver and Transmitter
PWM	Pulse Width Modulation
ICSP	In-Circuit Serial Programming
FTDI	Future Technology Devices international
USB	Universal Serial Bus
SST	Supersonic Transport
SAS	Statistical Analysis
RANS	Reynolds –averaged Navier-Strokes

NOMENCLATURE

A_p	Effective piston area, m^2
A_r	Rod area, m^2
C	Sound speed in water, m/s
C_v	Valve sizing coefficient of FCV, constant
D_p	Piston diameter, m
D_r	Rod diameter, m
F_{load}	Design force, N
K	Bulk modulus of elasticity, N/m^2
l	Pipe length, m
L	Cylinder stroke, m
P_{atm}	Atmospheric pressure, N/m^2
ΔP	Pressure change, N/m^2
SG	Specific gravity of iso VG 32 oil, m^3/kg
T_c	Closing time, s
Q_{FCV}	Flow rate, m^3/s
ρ	Density, kg/m^3

CHAPTER I

INTRODUTION

CHAPTER I

INTRODUTION

1.1 General

In 1959 contract assigned between Sudan and Egypt governments and decision making to transfer Halfa civilians to eastern Sudan beside Kasala city which is 600 km from Khartoum and it's called after that New Halfa. The Dam was constructed (1960-1964) to irrigate Halfa agricultural project for settlement all Halfa civilian. The dam consists of seven sluice gates and five spill ways to pass water in autumn to Atbara river stream and there are two compensating gates in Halfa canal, Fig. 1.1. Beside irrigation the Dam generates electricity. The capacity of reservoir is 1.3 billion m^3 and the reservoir area is 125 km^2.

Fig. 1.1 Khashm-Algirba Dam

The khashm-Algirba power station contains five hydropower units, two of them Kaplan turbine locate in the Atbara river stream and three are pump turbines placed in the Halfa canal gate, Fig. 1.2. All units in the station were rehabilitated and upgraded in 2004, since every Kaplan turbines were upgraded from 3.5 MW to 5.3 MW, and every unit of pump turbines upgraded from 1.8 MW to 2.4 MW, Fig. 1.3, the total capacity of the station is 17.8 MW. Also there were four compensating pump every one discharge

0.4 Mm^3/day. Also there were four diesel engines in station follow to the thermal company in 2011.

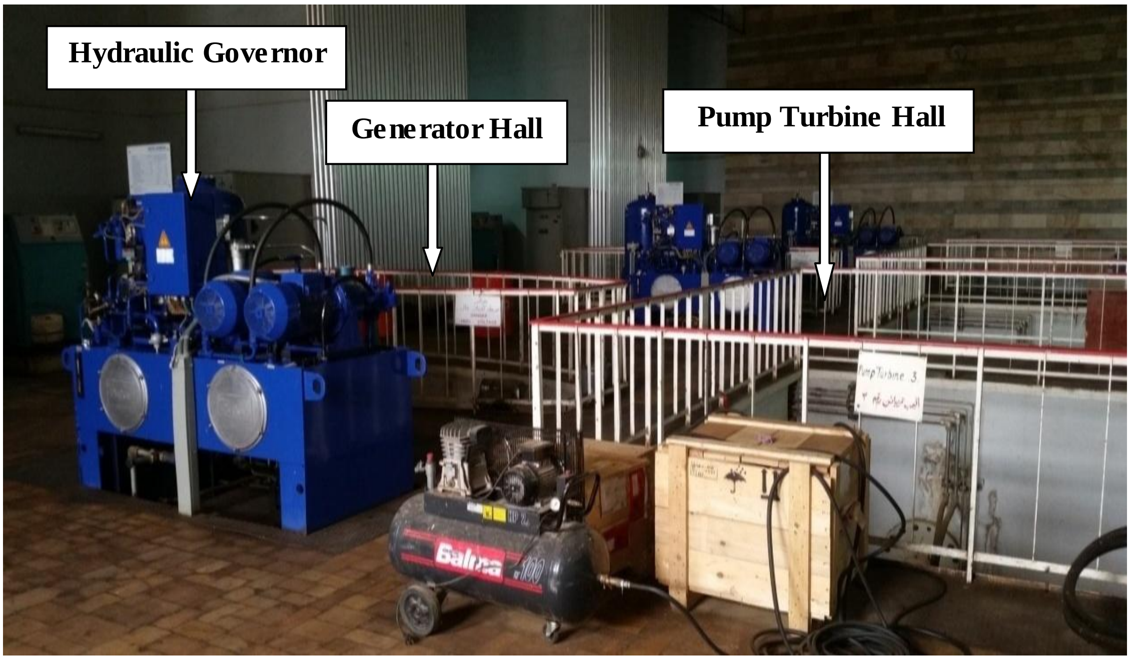


Fig. 1.2 Three Pump Turbines in Khashm-Algirba Hydropower Plant

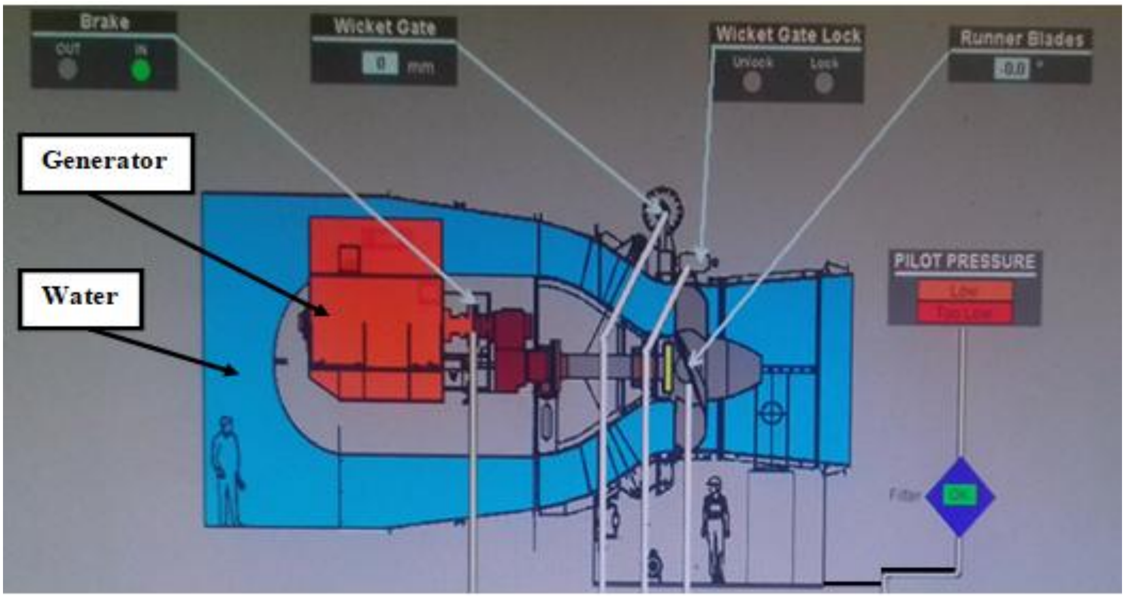


Fig. 1.3 Pump Turbine

Transformers:

A Transformer, Fig. 1.4, is a static electrical machine which transfers AC electrical power from one circuit to the other circuit at the constant frequency, but the voltage level can be altered that means voltage can be increased or decreased according to the requirement. The Station consists of two transformers each one 15 MVA with voltage level 6/66 KV.

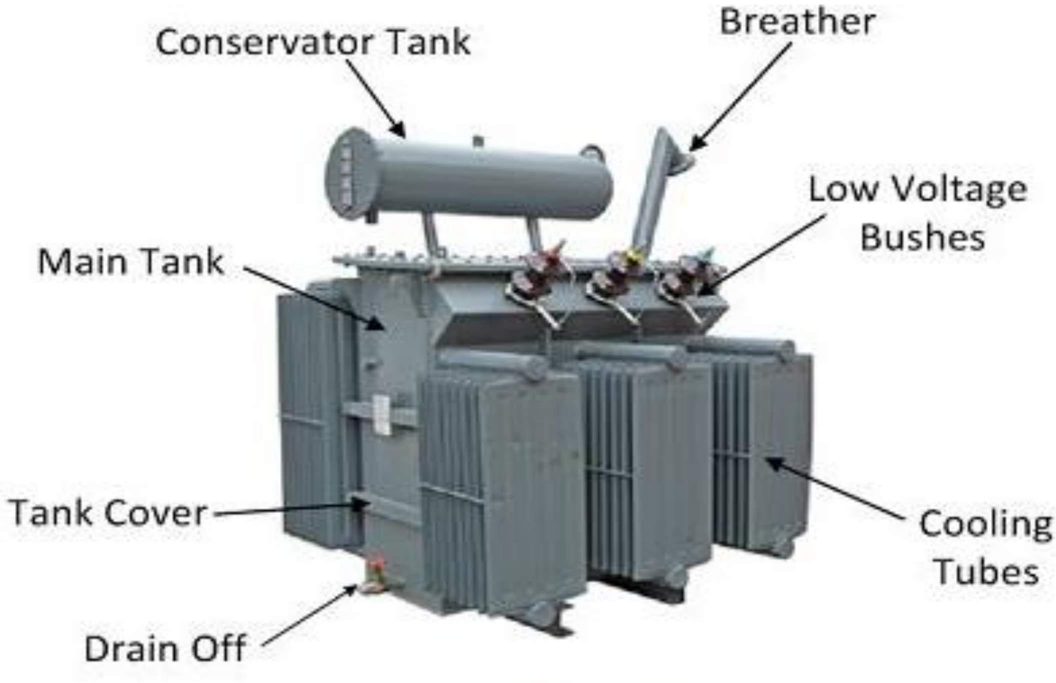


Fig. 1.4 Transformer

1.2 Overview

All hydropower turbines that connected to the grid have 50 Hz generators. The governor keeps each unit operating at its rated speed through a high pressure hydraulic system that operates wicket gates which control water flow through the turbine. When the load changes or disturbances in the power grid, the governor respond to open or close the Wicket gate (WG) to increasing or decreasing power output of the generating units to meet power demands and keep the frequency of the power grid at rated frequency. Governor operating characteristics will be determined from the electrical, mechanical, and hydraulic characteristics of the generator, turbine, and penstock. Older governors use mechanical speed sensing and control, interfaced to the hydraulic system to govern turbine speed. Newer systems incorporate electronic or digital speed sensing and controls with a hydraulic interface to the turbine governor.

The hydroelectric generation turbines have a protection mechanism to protect unit from runway speed by stop water flow through the turbine. When the signal sends to protection system to trip the unit due to any cause the WG will close to stop water flow. In case the signal for tripping unit is send to trip unit and the WG doesn’t response to close command due to hydraulic pressure loss or due to other anything the speed of the unit will increase, when reach to overspeed level two the power intake gate or downstream roller gate will close automatically to stop water flow through the turbine and then the unit will stop. In case there is only WG to stop water flow and the intake gate or downstream gate not existing or not connected to the automation system the speed of the runner will increase until reach overspeed level two and will continue increase until reach to the runway speed and a significant damage will occur in the turbine. The last one case existing in the series of pump turbines in Girba power station since there is no intake gate or inlet valve but there is downstream gate (DSG) in the units but not work automatically because it's not connected with the automation system.

The function of the DSG is to ensure a reliable protective system for bulb units providing, Fig. 1.5:

1. Protection against runaway conditions
2. Downstream isolation during unit dewatering
3. Sluice operation

This research will select controller and programming it to control DSRG for the pump turbine no. two in Girba power station to prevent the unit from runway speed.

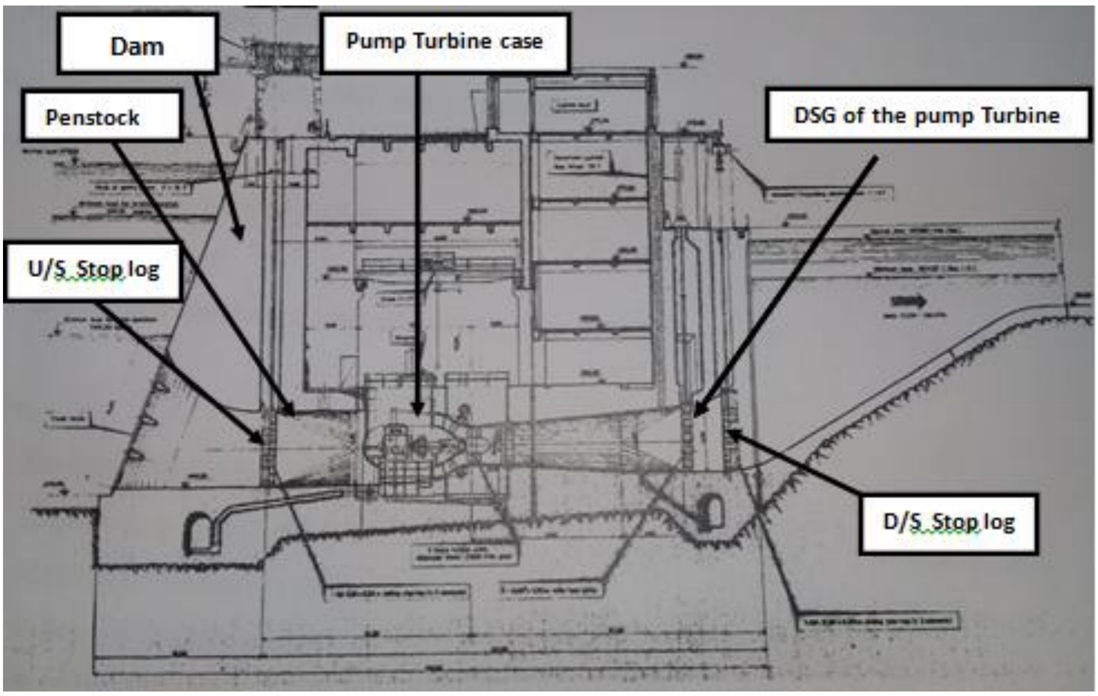


Fig. 1.5 Sectional view of the pump turbine

1.3 Problem Statement

Runaway speed in hydro-electrical generation unit is the speed that leads to total damage for the turbine. In case the WG servomotor failure to close the WG or the hydraulic pressure system of the pump turbine loss during the unit loaded the trip signal will send by automation system for tripping unit due to hydraulic system pressure alarm pressure too low and the Circuit Breaker (CB) will disconnect. the speed of the unit will increase until reach to the overspeed level two and will continue increase until reach to runaway speed, since there is nothing stop water flow through the turbine except the WG which operate by hydraulic pressure system and the DSG don't close automatically because it's not connected to the automation system and the manual control of the gate is far from operator panel and that result to significant damage in the turbine.

The causes of pressure system loss that leads to overspeed for the unit are:

1. Release pressure due to human error
2. Hose damage while unit running
3. Safety valve failure

1.4 Problem solution

The proposed solution for the WG servomotor failure to close the WG or the hydraulic pressure system loss is to design microcontroller which will process the input signals from pushbuttons or speed sensor and sent signal to DSG solenoid valve to close the DSG of the pump turbine and cut-off water.

1.5 Aims and objective

Main objective of this study is to design a microcontroller to control DSG of the Pump turbine.

1.6 Methodology:

The Arduino Uno, board, Direct Current (DC) motor, Driver, Infrared (IR) sensor, Liquid crystal Display (LCD), Pushbutton, Red and green Light Emitting Diode (LED), buzzer, wires are prepared.

The program of the system was written by Arduino C language in PC and then downloads to the Arduino Uno. All these devices are interfaced to the Arduino Uno and then the Arduino Uno board is connected to the power. The DC motor is driven and the speed reading by IR sensor and then sent to the Arduino Uno which process the signal and display the motor speed on the LCD screen, the rated speed considered at 750 rpm, the speed of the motor is gradually increased and when reach to the 1080rpm the red LED light which represent the overspeed level one, the speed continue rising, when it reach 1260 rpm the MCU sends signal to the DSG valve which represented by green LED. The model developed in laboratory kits by help from Engineer Johr.

1.7 Thesis Layout

Chapter 2: **Theory and Literature Review**: this chapter reviews the fundamental concepts and principles that the project relies on.

Chapter 3: **Methodology**: this chapter talks (1) about all equipments and tools used in the project implementation, mentioning brief description of the main features for each component; (2) about design and implementation in details. All hardware and software design steps are considered here including all physical requirements, algorithms, circuits block diagrams, flowcharts. This chapter describes implementation details of the system modules described in the design.

Chapter 4: **Results and Discussion**: this chapter constitutes the real work in order to achieve the project objectives.

Chapter 5: **Conclusion**: this chapter contains the conclusion and the recommendations for the future work.

CHAPTER II

THEORETICAL BACKGROUND

AND LITERATURE REVIEW

CHAPTER II

THEORETICAL BACKGROUND AND LITERATURE REVIEW

2.1 Background

Runaway speed is the maximum rotational speed to which a generating unit can be driven with an open circuit breaker and the available hydraulic and mechanical conditions. The overspeed protection system of the hydropower unit trips the unit when the speed reach to specific speed, the trip signal will send to the protection system to stop the unit by closing the wicket gate. In case the unit doesn't response to the stop command due to any fault in hydraulic governor, the speed will increase until limited by hydraulic conditions, Windage, and friction, and this may damage the unit (Kamaran Jamal Muhedeen, 2011).

So this system need to an improvement by enabling the interaction to stop the unit by closing the DSG to cut off water flow through turbine and then the unit will stop when the speed reach to the overspeed. There is pushbutton will connect with Automation system to close the DSG manually in emergency case.

2.2 Previous Studies

The study discussed a reliable overspeed protection system for turbines. This means reliable detection of overspeed and immediate closing of the trip valve in response. The VOITH CTo overspeed protection system provides both functions in a single device. An electronic monitoring unit detects overspeed and an electrohydraulic assembly actuates the trip valve immediately.

Advantages:

- Compact design
- Direct mounting onto the actuating mechanism of the trip valve on the turbine
- No space required in the control cabinet
- Short signal paths; on-site detection, evaluation and actuation
- Device insensitive to malfunctions
- Especially well-suited for turbines operating in a lower power range.
- Partial stroke test during operation
- Speed sensor optional, triply redundant

- Explosion-proof designs optional. (Voith Turbo Gmbh & Co. KG, 2012).

Abnormal operation of Francis pump turbines at runway resulting from guide vanes servomotor failure is standard case is investigated during transient analysis. Due to S-shape of pump-turbines characteristics in turbine mode. The operation at runaway usually leads to unstable behaviour characterized by a large period of oscillations typically between 8 to 20 seconds, corresponding to the so called hydro-mechanical mode. The period of the hydro mechanical mode depends on the rotating inertias the penstock water the analytical prediction of the period of the oscillations at runway provided by martin made during speed no load operation on the prototype of a Francis pump turbine. However during these tests, a lower period oscillation phenomenon was observed and described as the medium frequency unstable mode and it was noticed that the phenomenon was oscillating at a period close to the period of the first natural period.

The runaway speed range varies according to unit type. Typical runaway speeds for Kaplan turbines are 2.5 to 3 time's normal speed, whereas with Impulse and Francis turbines they are rarely over twice normal. Runaway speed values are measured during the steady-state model tests of the turbine. In operation, a unit cannot exceed the runaway speed for a particular net head and gate opening. Turbo-generating machinery must be designed to be strong enough to withstand runaway speed, but with a highly reduced safety factor. However, as mentioned in Section 5.3, the prototype speed may exceed the steady-state runaway speed for a short duration in transient states since the instantaneous head is governed by water hammer effects. Therefore, the turbine steady state characteristic curves have to be extended to obtain the transient runaway speed. The same concepts and observations about runaway condition hold for both isolated turbines and the turbines originally interconnect (Qinfen Katherine Zhang, 2009).

Runaway speed is the maximum rotational speed to which a generating unit can be driven with an open circuit breaker and the available hydraulic and mechanical conditions. The term usually refers to a fixed wicket gate opening, and in the case of Kaplan turbines, a fixed blade angle. During a load rejection, the water column continues to provide energy to the turbine runner. Since this energy can no longer be converted into electrical energy, a portion is mechanically stored and the rest is dissipated in turbulence before being discharged from the turbine. The energy is stored via increased angular momentum of the turbine runner, shaft, and generator rotor. The total amount of energy that can be stored is a function of the rotating inertia, or WK2, and the increase in rotational speed. If the wicket gates do not move to a closed position, the speed will increase until limited by hydraulic conditions, windage, and friction. The hydraulic conditions include the available head, the turbine's performance characteristics such as off design efficiency, and cavitation, which can reduce the efficiency of the energy transfer. Windage refers to air resistance, mostly in the generator, and friction refers to mechanical "sliding" friction. Ultimately, the decreased amount of fluid energy that can be transferred from the water column is balanced by the increased Windage and friction, at which point runaway speed is achieved. The higher the head, the larger the wicket gate opening, or the flatter the blades on a Kaplan turbine, the higher the runaway speed. For this latter reason, the blades on Kaplan turbines are often designed to tilt to their steepest position on loss of governor control. Francis and Kaplan turbines have different runaway speed characteristics. Francis turbines typically have less WK2 than Kaplan and, therefore, achieve runaway speed faster. At runaway speed, Francis turbines tend to "choke" the flow, reducing the discharge. Kaplan, on the other hand, tend to increase the flow with increasing speed at a given gate and blade angle. On Kaplan turbines, once runaway speed is achieved if load is rejected, the gates do not move, and the blades are at the proper cam position for the gate opening and head and do not move. If the blades move to any other position without moving the gates, it is referred to as off-cam runaway speed. Also with Kaplan, maximum runaway discharge is when the blades are full steep. If the wicket gates are free to move and the unit is under governor control, a shutdown sequence is initiated upon load rejection. However, since the wicket gates take a finite time to close, a transient increase in synchronous speed, known as over speed, is achieved. In order to limit this over speed, the wicket gates should close as quickly as possible. However, the faster they close, the higher the pressure transient of water hammer that is sent back upstream. For this reason the rate of closure, called the gate-timing element, is a compromise between over speed and water hammer. The peak over speed can be reduced by increasing the inertia of the rotating parts. Normally, the maximum design over speed is 150 % of synchronous speed (Kamaran Jamal Muhedeen, 2011).

CHAPTER III

METHODOLOY

CHAPTER III

METHODOLOY

3. Equipment's and Tools

3.1 Introduction

The hydraulic turbine is the machine that used to convert the hydraulic energy that stored in the water flow into electrical energy, its design to rotate at constant speed and there is ability to control the speed to remain within rated speed, in case the speed increased the unit must be stop by signal from control system. There are some of equipments and tools used to design control system of the DSG of the pump turbine to stop water flow through machine and stop unit to protect it from runway speed.

3.2 Hardware component and description

3.2.1 Microcontroller

There are some criteria for choosing a microcontroller:

1) The first and foremost criterion for choosing a microcontroller is that it must meet the task at hand efficiently and cost effectively. In analyzing the needs of a microcontroller-based project, it is seen whether an 8-bit, 16-bit or 32-bit microcontroller can best handle the computing needs of the task most effectively. Among the other considerations in this category are:

(a) Speed – What is the highest speed that the microcontroller supports?

(b) Packaging – Does it come in 14-pin DIP (dual inline package) or a QFP (quad flat package), or some other packaging format? This is important in terms of space, assembling, and prototyping the end product.

(c) Power consumption – This is especially critical for battery-powered products.

(d) The number of input output (I/O) pins and the timer on the chip.

(f) How easy it is to upgrade to higher – performance or lower consumption versions.

(g) Cost per unit – this is important in terms of the final cost of the product in which a microcontroller is used.

2) The second criterion in choosing a microcontroller is how easy it is to develop products around it. Key considerations include the availability of an assembler, debugger, a code –efficient compiler, technical support.

3) The third criterion in choosing a microcontroller is its ready availability in needed quantities both now and in the future. Currently of the leading 8-bit microcontrollers (Ewetumo Theophilus, 2012).

The type of the MCU that used to execute this research is the Arduino Uno (it's the Board Based on Atmega 328/P)

The Atmel® picoPower® ATmega328/P, Fig. 3.1 is a low-power CMOS 8-bit microcontroller based on the AVR® enhanced RISC architecture. By executing powerful instructions in a single clock cycle. See summary of configuration for ATmega328 Table 3.1

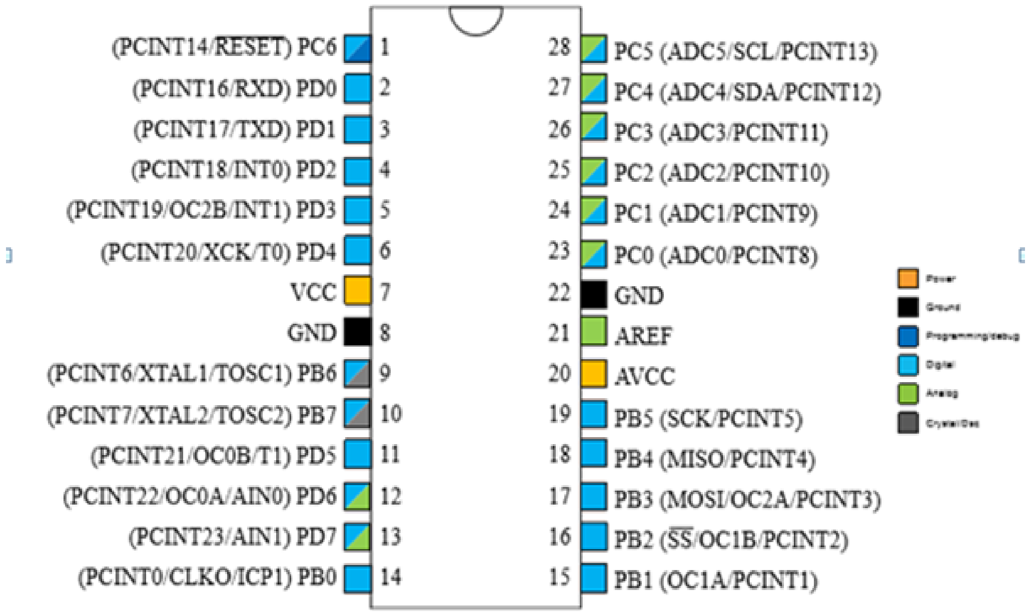


Fig. 3.1 Pin-out 28-Pin PDIP

Atmega 328/P Pin Descriptions:

1. VCC Digital supply voltage.

2. GND Ground.

3. Port B (PB[7:0]) XTAL1/XTAL2/TOSC1/TOSC2 Port B is an 8-bit bi-directional I/O port.

4. Port C (PC[5:0]) Port C is a 7-bit bi-directional I/O port.

5. PC6/RESET If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin.

6. Port D (PD[7:0]) Port D is an 8-bit bi-directional I/O port.

7. AVCC AVCC is the supply voltage pin for the A/D Converter, PC[3:0], and PE[3:2]. It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter. Note that PC[6:4] use digital supply voltage, VCC. 5.2.

8. AREF: AREF is the analog reference pin for the A/D Converter.

9. ADC [7:6] serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels (Atmel Corporation, 2016).

Table 3.1 Atmega328 Configuration Summary

Features	ATmega328/P
Pin Count	28/32
Flash (Bytes)	32K
SRAM (Bytes)	2K
EEPROM (Bytes)	1K
General Purpose I/O Lines	23
SPI	2
TWI(I^2C)	1
USART	1
ADC	10-bit 15kSPS
ADC Channels	8
8-bit Timer/Counters	2
16-bit Timer/Counters	1

ARDIUNO UNO Board:

The Arduino Uno is a microcontroller board Fig. 3.2 based on the ATmega 328, It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The technical specification of the Arduino Uno explained in Table 3.2. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards.

Table 3.2 Arduino Uno Technical Specification

Microcontroller	ATmega328
Operating Voltage	5V Input
Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14(of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2KB
EEPROM	1KB
Clock Speed	16MHz

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. 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.

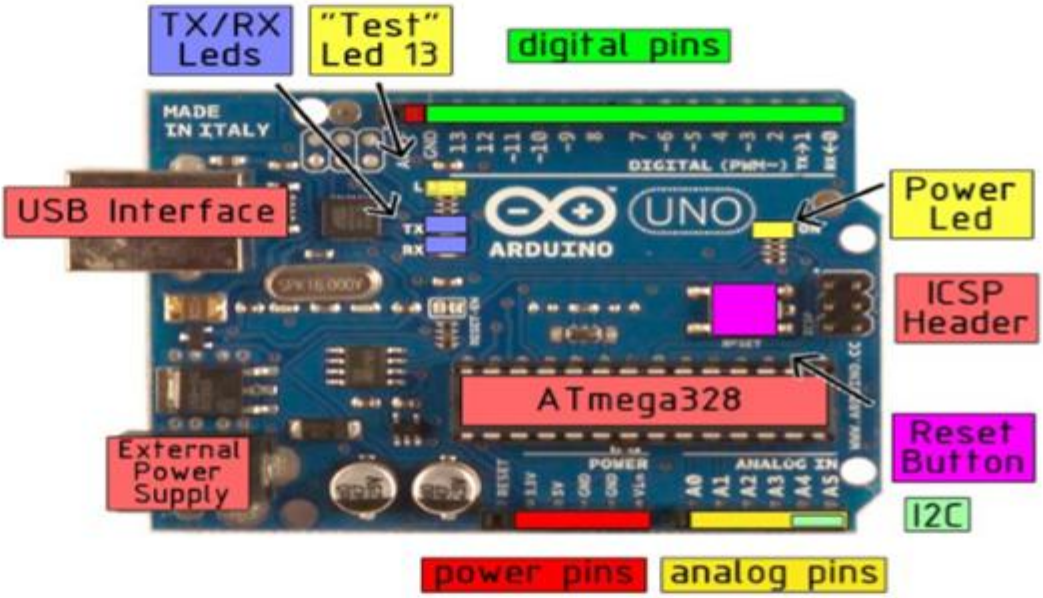


Fig. 3.2 Arduino Uno Board

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. 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.

The power pins are as follow

VIN. The input voltage to the Arduino board when it's using an external power source.

5V. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.

3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

GND. Ground pins

Memory

The Atmega 328 has 32 KB of flash memory for storing code (of which 0,5 KB is used for the bootloader); It has also 2 KB of SRAM and 1 KB of EEPROM

Input and output

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode (), digitalWrite (), and digitalRead () 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 ohms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. T These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write() function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analog Reference () function. Additionally, some pins have specialized functionality:

- I2C: 4 (SDA) and 5 (SCL). Support I2C (TWI) communication using the Wire library.

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analog Reference ().
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Physical Characteristic:

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 (0.16"), not an even multiple of the 100 mil spacing of the other pins.

3.2.2 Proximity Sensors:

Proximity Sensors are available in models using high-frequency oscillation to detect ferrous and non-ferrous metal objects and in capacitive models to detect non-metal objects. Models are available with environment resistance, heat resistance, resistance to chemicals, and resistance to water.

Proximity Sensor includes all sensors that perform non-contact detection in comparison to sensors, such as limit switches, that detect objects by physically contacting them. Proximity Sensors convert information on the movement or presence of an object into an electrical signal. There are three types of detection systems that do this conversion: systems that use the eddy currents that are generated in metallic sensing objects by electromagnetic induction, systems that detect changes in electrical capacity when approaching the sensing object, and systems that use magnets and reed switches.

The Japanese Industrial Standards (JIS) gives the generic name "proximity switch" to all sensors that provide non-contact detection of target objects that are close by or within the general vicinity of the sensor, and classifies them as inductive, capacitive, ultrasonic, photoelectric and magnetic.

This Technical Explanation defines all inductive sensors that are used for detecting metallic objects, capacitive sensors that are used for detecting metallic or non-metallic objects, and sensors that utilize magnetic DC fields as Proximity Sensors.

Features

1. Proximity Sensors detect an object without touching it, and they therefore do not cause abrasion or damage to the object.
2. No contacts are used for output, so the Sensor has a longer service life (excluding sensors that use magnets).
3. Unlike optical detection methods, Proximity Sensors are suitable for use in locations where water or oil is used.
4. Proximity Sensors provide high-speed response, compared with switches that require physical contact.
5. Proximity Sensors can be used in a wide temperature range from (-40 to 200°C)
6. Proximity Sensors are not affected by the object's surface color.
7. Unlike switches, which rely on physical contact, Proximity Sensors are affected by ambient temperatures, surrounding objects, and other Sensors.

Both Inductive and Capacitive Proximity Sensors are affected by interaction with other Sensors. Because of this, care must be taken when installing them to prevent mutual interference.

8. There are Two-wire Sensors.

The power line and signal line are combined. If only the power line is wired, internal elements may be damaged.

IR (Infrared) Sensors

The basic concept of IR Sensor Fig. 3.3 obstacle detection is to transmit the IR signal (radiation) in a direction and a signal is received at the IR receiver when the IR radiation bounces back from a surface of the object.

The Sensor is the front end device which comes directly in contact with the quantity being measured. In microcontroller based TURBINE, the choice of transducer or sensor to measure the speed, the speed of the turbine is very critical. So we require speed sensor:

The transducer should have following properties :

1. Accuracy
2. High Output
3. Repeatability
4. Long term stability
5. High Input Impedance
6. vi. Linearity
7. Self-Heating
8. Temperature Compensation (Varuninder Singh, 2006).

Principle of operation

The IR transmitter sends an infrared signal that, in case of a reflecting surface (e.g. white color), bounces off in some directions including that of the IR receiver that captures the signal detecting the object. When the surface is absorbent (e.g. black color) the IR signal isn't reflected and the object cannot be detected by the sensor, Fig. 3.4. This result would occur even if the object is absent. In this bank vault security system design, in order to make a bidirectional counter for restricting the number of people coming close to the vault door, the two IR sensors were used. It is a 0 to 9 counter in which the first sensor (IR-a) is used to count ingoing people, the second (IR-b) those outgoing. (Gerald Felix and Jiang De Ning, 2015).

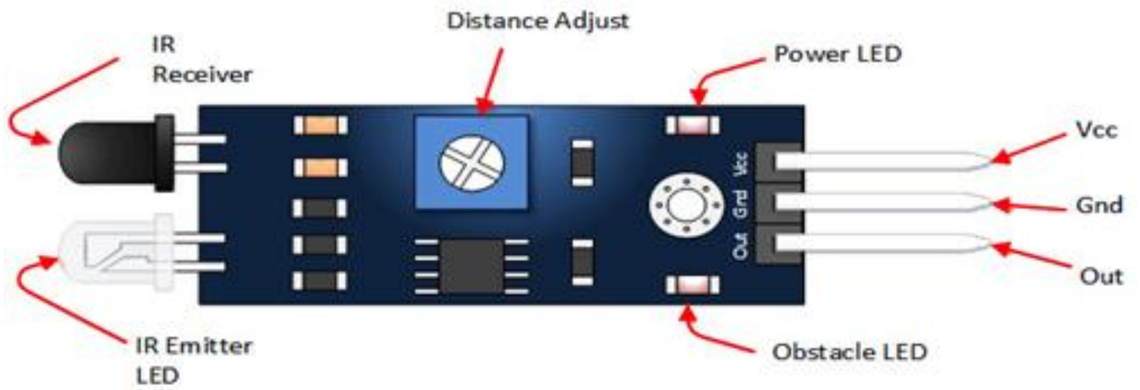


Fig. 3.3 IR Sensor

Features

There is an obstacle, the green indicator light on the circuit board

Digital output signal

Detection distance: 2 ~ 30cm

Detection angle: 35 ° Degree

Comparator chip: LM393

Adjustable detection distance range via potentiometer

Clockwise (CW): Increase detection distance

Counter-clockwise (CCW): Reduce detection distance (Gerald Felix and Jiang De Ning, 2015).

Specifications

Working voltage: 3 - 5V DC

Output type: Digital switching output (0 and 1)

3mm screw holes for easy mounting

Board size: 3.2 x 1.4cm (Gerald Felix and Jiang DeNing, 2015).

The table 3.3 shows the IR sensor pin description

Table 3.3 Summary of pin description for IR sensor

Pin	Description
Vcc	3.3 to 5 Vdc Supply Input
Gnd	Ground Input
Out	Output that goes low when obstacle is in range
Power LED	Illuminates when power is applied
Obstacle LED	Illuminates when obstacle is detected
Distance Adjust	Adjust detection distance. CCW decreases distance. CW increases distance.
IR Emitter	Infrared emitter LED
IR Receiver	Infrared receiver that receives signal transmitted by Infrared emitter.

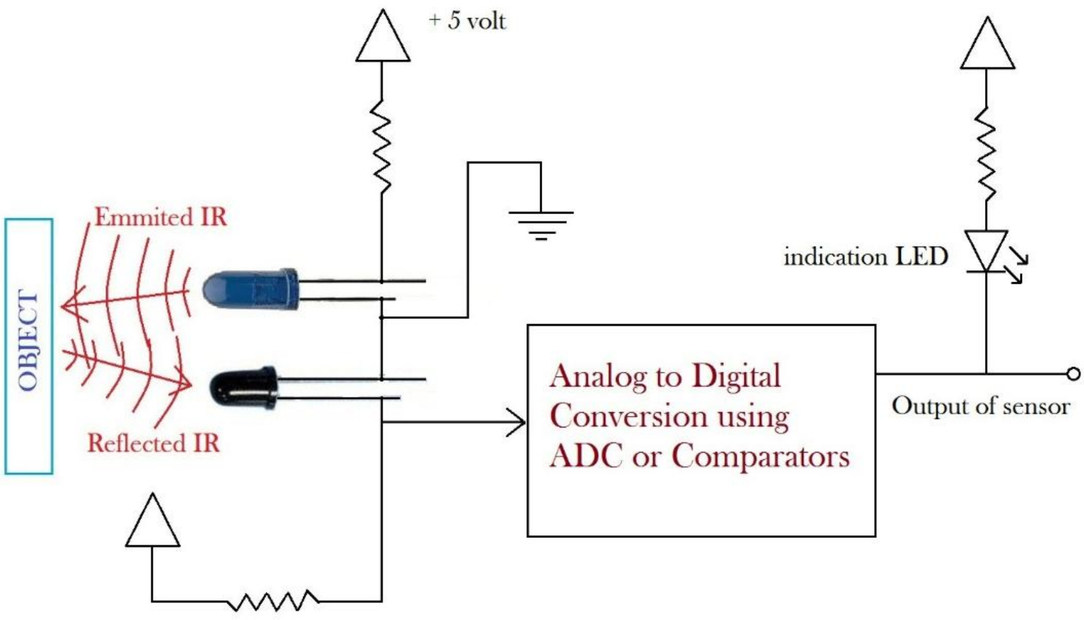


Fig. 3.4 Schematic for IR Sensor

3.2.3 LED

A light-emitting diode (LED) Fig. 3.5 is a two-lead semiconductor light source. It is a P–N junction diode. When a suitable voltage is applied to the leads, electrons are able to recombine with electron holes within the device, releasing energy in the form of photons. A P-N junction can convert the absorbed light energy into its proportional electric current. The same process is reversed here; this phenomenon is generally called electroluminescence, which can be defined as the emission of light from a semi-conductor under the influence of an electric field. The charge carriers recombine in a forward P-N junction as the electrons cross from the N-region and recombine with the holes existing in the P-region. Free electrons are in the conduction band of energy levels, while holes are in the valence energy. Thus the energy level of the holes will be lesser than the energy levels of the electrons. Some part of the energy must be dissipated in order to recombine the electrons and the holes. This energy is emitted in the form of heat and light. And the color of the light (corresponding to the energy of the photon) is determined by the energy band gap of the semiconductor, the schematic of LED Fig. 3.6

An LED is often small in area (less than 1 mm2) and integrated optical components may be used to shape its radiation pattern. (Shwetha K.S et al, 2015).

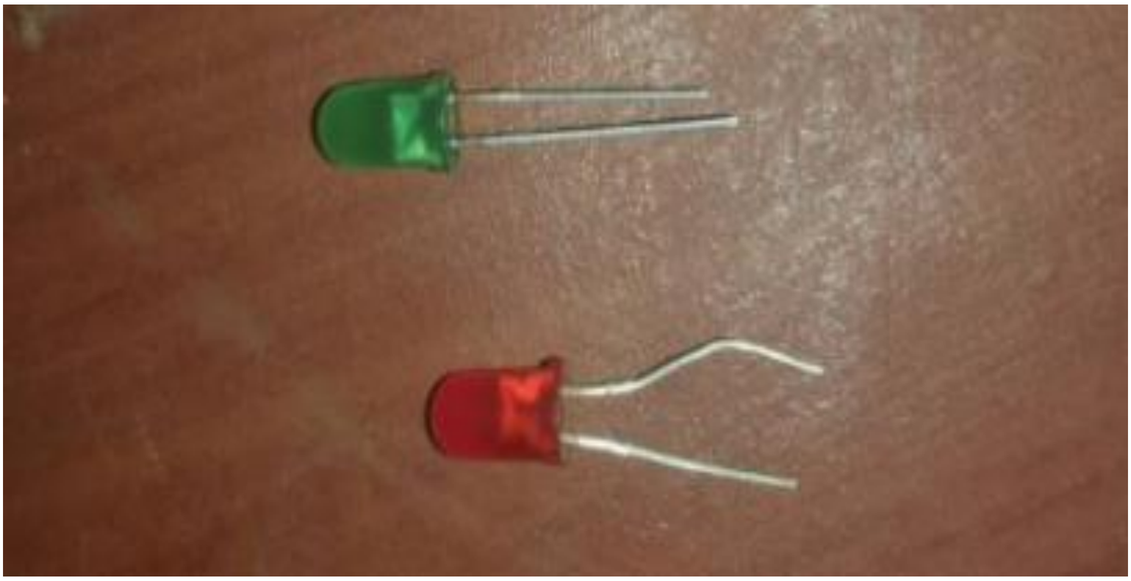


Fig. 3.5 LED

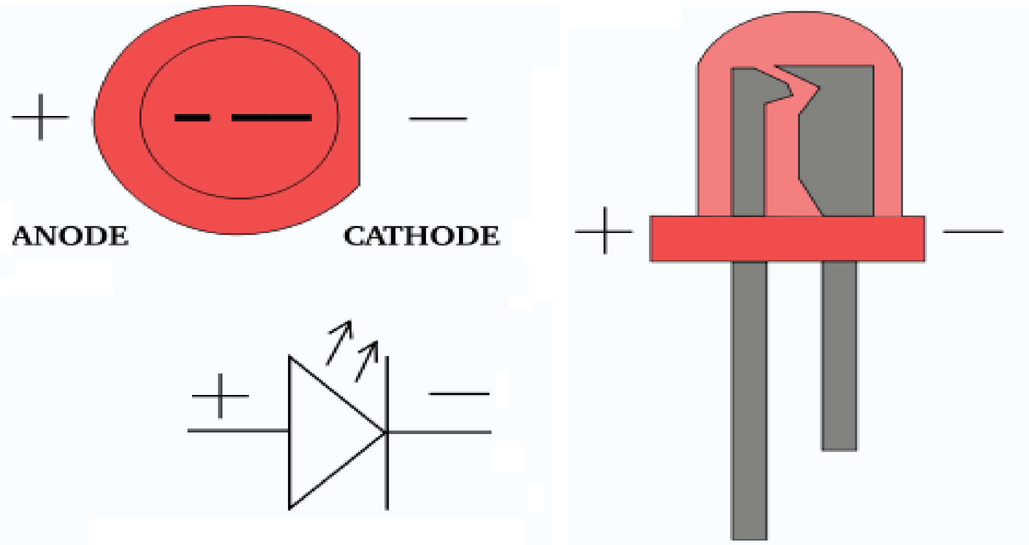


Fig. 3.6 Schematic for LED

3.2.4 Buzzer

A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric Fig. 3.7. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke the schematic of buzzer is shown in Fig.3.8.

Fig. 3.7 Buzzer

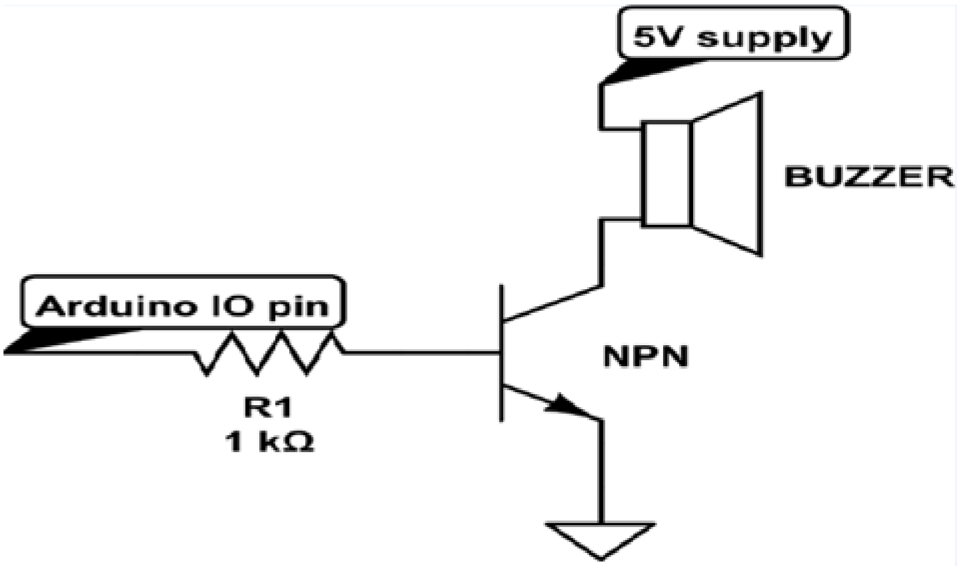


Fig. 3.8 Schematic for buzzer

3.2.5 LCD

LCD stands for Liquid Crystal Display. An LCD is a passive device.16×2 Character LCD is a very basic and low cost LCD module which is commonly used in electronic products and projects. 16×2 means it contains 2 rows that can display 16 characters. Its other variants such as 16×1 and 16×4 are also available in the market.

Almost all LCDs have a strong light source built in behind a glass panel (which contains the liquid crystal), this ensures that the areas of light and dark on the screen have good contrast. Displays with no backlights that rely solely on the light incident on the LCD panel cannot be used in low light conditions.

Pin Description:

The most commonly used LCDs found in the market today are 1 Line, 2 Line or 4 Line LCDs which have only 1 controller and support at most of 80 characters, where as LCDs supporting more than 80 characters make use of 2 HD44780 controllers.

Most LCDs with one controller have 14 pins or 16 pins (two extra pins are for back-light LED connections) while LCDs with two controllers have two more pins to enable the additional controller. We will focus on LCDs with one controller Fig. 3.9, in this article since it is one of the commonly used models in the market. Likewise, the operations are applicable for two controllers LCD as well. Pin description is shown in the table 3.4.

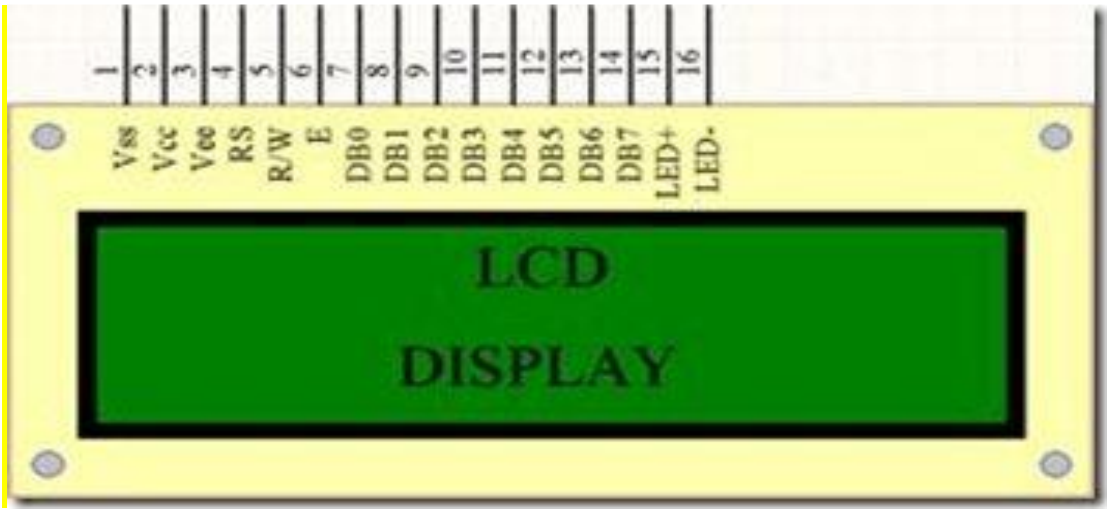


Fig. 3.9 Character LCD common pins diagram

Table 3.4 Summary of Pin Description for LCD Display

Pin No.	Name	Description
1	Vss	Power Supply (GND)
2	Vcc	Power Supply (+5v)
3	Vee	Contrast Adjust
4	RS	0 = Instruction Input , 1 = Data Input
5	R/W	0 = Write to LCD module, 1 = Read from LCD module
6	EN	Enable Signal
7	DB0	Data bus line 0 (LSB)
8	DB1	Data bus line 1
9	DB2	Data bus line 2
10	DB3	Data bus line 3
11	DB4	Data bus line 4
12	DB5	Data bus line 5
13	DB6	Data bus line 6
14	DB7	Data bus line 7 Most significant bit (MSB)
15	LED+	Back light supply (+V)
16	LED-	Back light supply (GND)

Character LCD pins with one controller

The HD44780 standard requires 3 control lines as well as either 4 or 8 I/O lines for the data bus. The user may select whether the LCD is to operate with a 4-bit data bus or an 8-bit data bus. If a 4-bit data bus is used the LCD will require a total of 7 data lines (3 control lines plus the 4 lines for the data bus). If an 8-bit data bus is used the LCD will require a total of 11 data lines (3 control lines plus the 8 lines for the data bus).

The three control lines are referred to as **EN**, **RS**, and **RW**.

The **EN** line is called "Enable." This control line is used to tell the LCD that you are sending in data. To send data to the LCD, your program should make sure this line is low (0) and then set the other two control lines and/or put data on the data bus. When the other lines are completely ready, bring **EN** high (1) and wait for the minimum amount of time required by the LCD datasheet (this varies from LCD to LCD), and end by bringing it low (0) again.

The **RS** line is the "Register Select" line. When RS is low (0), the data is to be treated as a command or special instruction (such as clear screen, position cursor, etc.). When RS is high (1), the data being sent is text data which should be displayed on the screen. For example, to display the letter "T" on the screen you would set RS high.

The **RW** line is the "Read/Write" control line. When RW is low (0), the information on the data bus is being written to the LCD. When RW is high (1), the program is effectively querying (or reading) the LCD. Only one instruction ("Get LCD status") is a read command. All others are write commands–so RW will almost always be low.

The data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as **DB0**, **DB1**, **DB2**, **DB3**, **DB4**, **DB5**, **DB6**, and **DB7**.

Finally, the power supply pins for the backlight – **LED+** and **LED-**. Some LCD modules come without the backlight. In that case, these pins are not found or are left disconnected. The recommended voltage for LED+ is 4.2V and LED- should be connected to ground (GND).

3.2.6 Pushbutton:

Push Button Switches consist of a simple electric switch mechanism which controls some aspect of a machine or a process Fig. 3.10. Buttons are typically made out of hard material such as plastic or metal. The surface is usually shaped to accommodate the human finger or hand, so the electronic switch can be easily depressed or pushed. Also, most Push Button Switches are also known as biased switches. A biased switch, can be also considered what we call a "momentary switch" where the user will push-for "on" or push-for "off" type. This is also known as a push-to-make (SPST Momentary) or push-to break (SPST Momentary) mechanism.

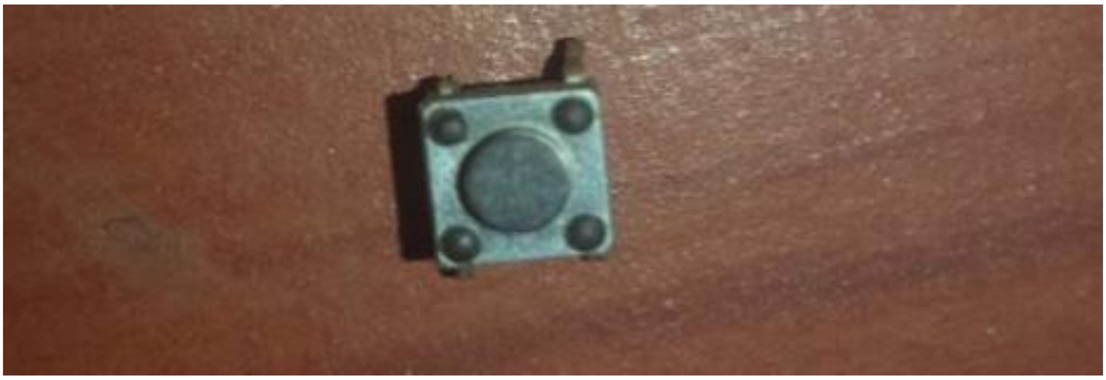


Fig. 3.10 Pushbutton

Switches with the "push-to-make" (normally-open or NO) mechanism are a type of push button electrical switch that operates by the switch making contact with the electronic system when the button is pressed and breaks the current process when the button is released. An example of this is a keyboard button.

A "push-to-break" (or normally-closed or NC) electronic switch, on the other hand, breaks contact when the button is pressed and makes contact when it is released.

Pushbutton Switches Applications

Typical applications include: Calculators, push-button telephones, kitchen appliances, magnetic locks, and various other mechanical and electronic devices, home and commercial.

3.2.7 DC motor:

Its operation is based on the principle that when a current carrying conductor is placed in a magnetic field, the conductor experiences a mechanical force. The direction of the force is given by Fleming’s left hand rule, DC motor is shown in Fig. 3.11.

Fig. 3.11 DC motor

3.3 Design and Implementation

3.3.1 Overview:

In this section the model electronic circuit will be explained the design to know how the system works and just to sensing the speed of the turbine and put two limits of speed. Limit one when activated the signal will send to the BUZZER and red LED to make alarm. Limit two when activated the signal will sent to the green LED. The last signal represents the signal of the shut off valve of the water in the pump turbine. And the limits of speed put due to the design of turbine.

For the design and development of the system, the methodology used involves the software and hardware implementation. The actual implementation of the system involves the following steps:

1. System Definition: Broad definition of system hardware including micro-controller and its interface with display, IR sensor, LEDs, buzzer and DC motor
2. Circuit Design: Selection of microcontroller and other interfacing devices, as per system definition. Design of hardware circuit and its testing on laboratory kits with some simple microcontroller software routines.

3. PCB Design and Fabrication: Generation of schematic diagrams and the production of circuit board layout data for the procurement of the circuit board.
4. Hardware Modifications: Making any hardware changes found necessary after the initial hardware tests, to produce a revised circuit board schematic diagram and layout.
5. Software Design: Developing algorithm for the system, allocating memory blocks as per functionality, coding and testing.
6. Integration and Final Testing: Integrating the entire hardware and soft-ware modules and its final testing for operation.

Thus the complete design is divided into two parts:

1. Hardware Implementation.

2. Software Implementation.

3.3.2 Hardware Details of the system

The block diagram of microcontroller based overspeed controller in the pump turbine is shown in Fig. 3.12.

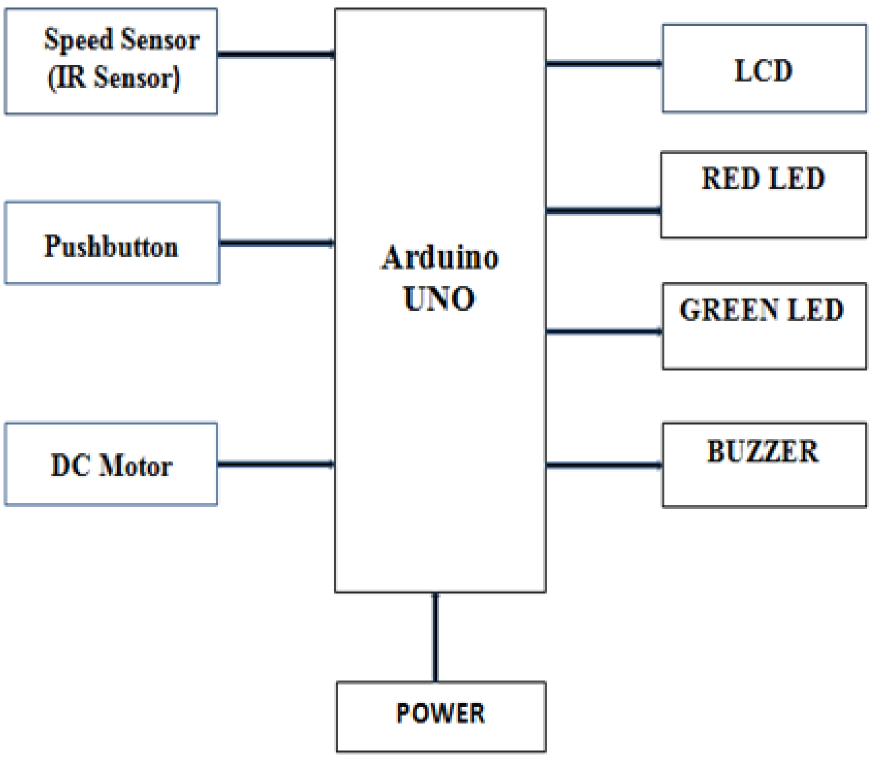


Fig. 3.12 The block diagram of the system

3.3.3 Hardware Implementation

It contains the design specifications. The design of hardware consists of the selection of system components as per the requirement, the details of subsystems that are required.

A typical LCD consists of 16 pins that control various features of the screen. A table that shows the pins and describes each function can be seen in Table 4.1 below. The Arduino microcontroller can output voltages of either 5 V or 3.3 V, so the LCD can be powered by wiring VSS and VDD to the ground and 5 V pins on the microcontroller. It is possible to adjust the contrast of the screen by wiring a variable resistor to V0 located at pin 3 on the screen. The RS, R/W, and E pins are wired to pins 12, ground, and 11 respectively on the Arduino. The LCD screen can operate in both 8-but mode and 4-bit. For this application note only 4-bit mode will be discussed, as it requires fewer pins and is generally easier to use.

3.3.4 Interface LCD with Arduino Uno

To interface with the LCD in 4-bit mode the Arduino only needs to be connected to pins DB4-DB7 Fig. 3.13, which will connected to digital output pins 5-2 respectively. Pins 15 and 16 on the LCD screen are used to power a backlight in the screen. This makes text displayed in the screen easier to read in poorly lit environments and is optional. In order to power the backlight pin 15 should be connected to ground while pin 16 should be connected to the 5 V output of the Arduino. To power the Arduino a 9 V battery can be connected to the Vin and ground pins on the Arduino. If such power source is available the Arduino can be powered by using its USB connection with a computer. Figure 3.13 shows what the final wiring scheme should look like after all connections are made.

3.3.5 Connect Detection Module to Arduino (Henry's Bench)

Fig. 3.14 explain general connect of detection module IR sensor with Arduino Uno.

The speed is very important parameter that needs to be monitored continuously in the turbines. Speed can be displayed in terms rpm, Fig. 3.15 is shown interconnect Arduino with IR sensor.

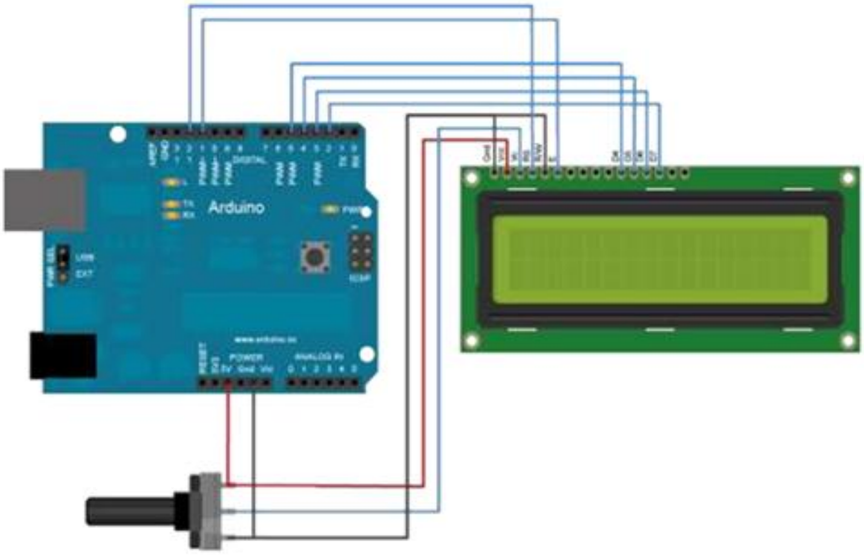


Fig. 3.13 wiring schematic for LCD Arduino interfacing

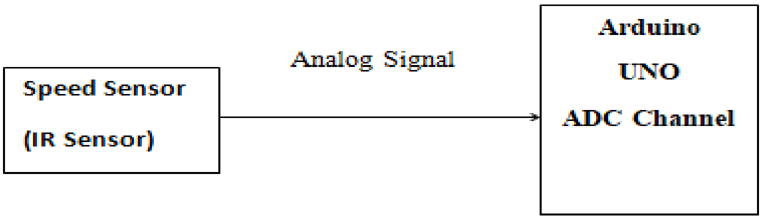


Fig. 3.14 IR Sensor interface to Arduino Uno MCU

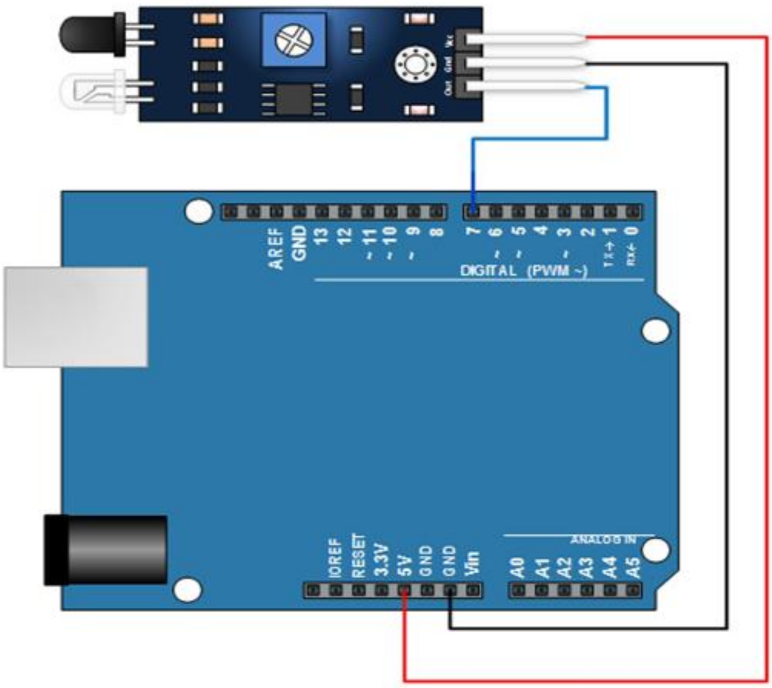


Fig. 3.15 Interface Arduino with IR Sensor

The speed is very important parameter that needs to be monitored continuously in the turbines. Speed can be displayed in terms rpm, Fig. 3.15 is shown interconnect Arduino with IR sensor.

3.3.6 Interfacing the Arduino UNO to various components:

The connection of the Arduino Uno with all devices is explained in detailed in Table 3.5

Table 3.5 Arduino Uno Pin Connection with devices

Arduino Uno Pins	Device
Pin(2)	IR Sensor
Pin(-3)- Pin(4)- Pin(-5)- Pin(-6)- Pin(7)- Pin(8) - (5V) – GRD	LCD
Pin(-9)	Pushbutton
Pin (-10)	Over speed Alarm (LED)
Pin(-11)	DC Motor
Pin(12)	DSG Close Signal
A0	Pot.

The connection of microcontroller with various components of DSG control for the turbine is shown below in detail in fig. 3.16.

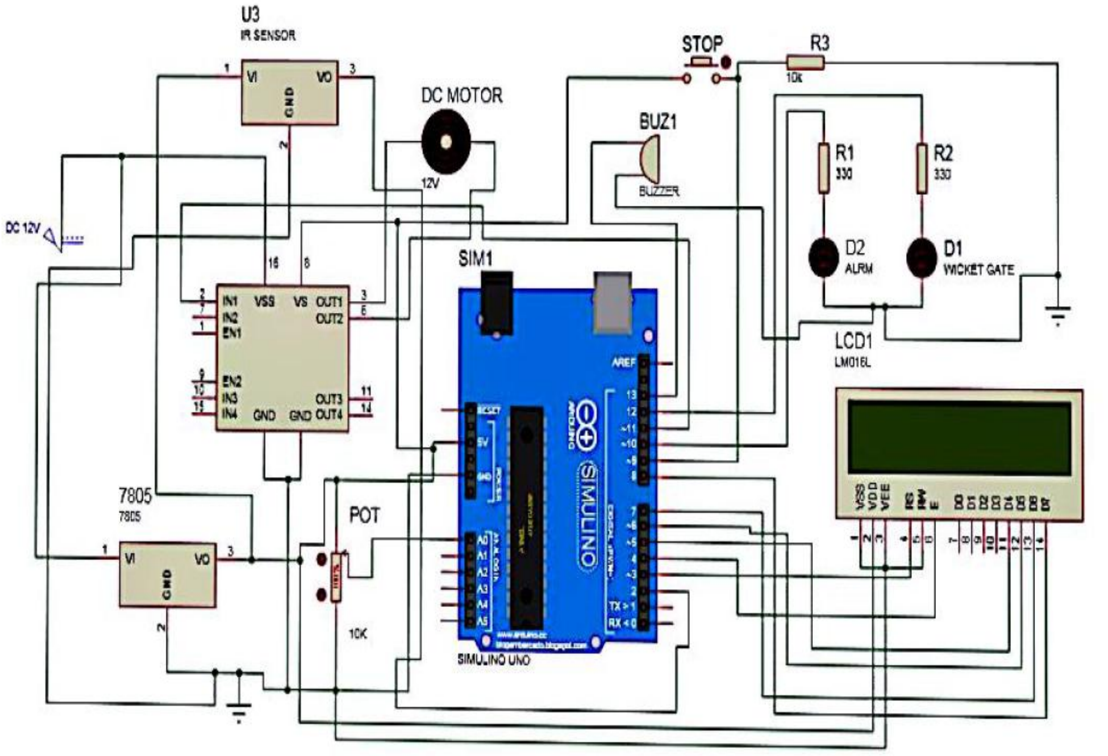


Fig. 3.16 Interfacing of Arduino Uno with Various components

3.3.7 Wiring all Components with Arduino Uno:

The wiring of the system and hardware schematics for the PCB layout, Fig. 3.17. design of the circuit and its testing has been carried out.

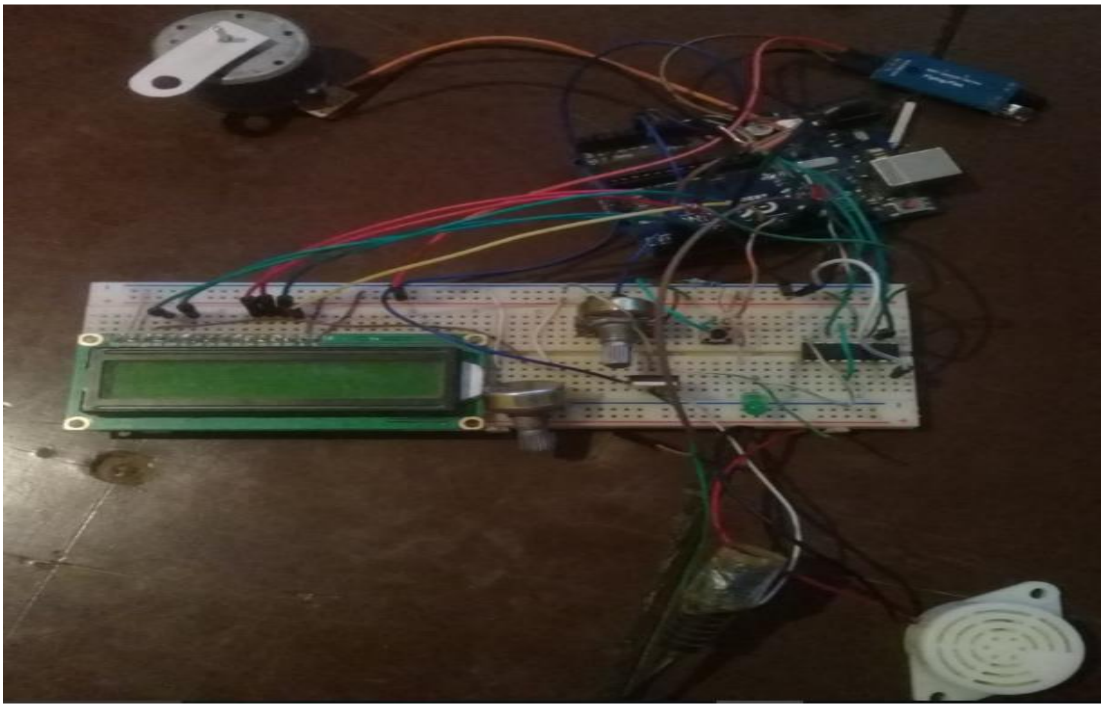


Fig. 3.17 Schematic Hardware Design

3.3.8 Operation of DSG control system:

The DSG control system is based on the Arduino Uno MCU was developed for PT2 to protect it from run way speed.

The driver interfaced to make switching voltage of 9 volts required by the unipolar motor from the voltage of work of the microcontroller.

In this mechanism, the DSG servo valve receives a signal from the MCU in case the speed of the turbine reach to overspeed level two or when press pushbutton, then the DSG move down to stop water flow through the turbine and protect it from runway speed.

3.4 Software design

3.4.1 Overview

Software design includes developing algorithm for the system, allocating memory blocks as per functionality, writing the separate routines for different interfacing devices and testing them on the designed hardware. Interfacing the microcontroller with LCD, pushbutton, IR sensor, LED, Driver, Buzzer and DC motor has been carried out. The control program is written in C language. The software is able to show the real time values from the analog channels for immediate analysis. For designing the software for this work, the flow of software between the hardware components is to be understood first.

3.4.2 Software Details of the system:

Software is an integral part of any control system, it interacts with hardware to carry out different functions which are responsible for the control of parameters. In the given problem the software can be divided into the following subparts:

- To assign different ports (pins) of MCU to different components of the system.
- To display different input values.
- To accept the set point input value of speed of the turbine
- To compare the real time values to input values and perform necessary control action.
- To display the results, Atmega328P MCU is the main brain of the D.S.R.G control of the pump turbine, so the algorithm inside it must be capable to control and perform all operations including reading the sensor outputs, processing these readings.

3.4.3 Flowcharts for Software Development:

Flowcharts have been developed in this section depicting step by step development of the software which issue instructions to various components of the hardware thereby making monitoring and control of parameters efficient and automatic, Fig. 3.18.

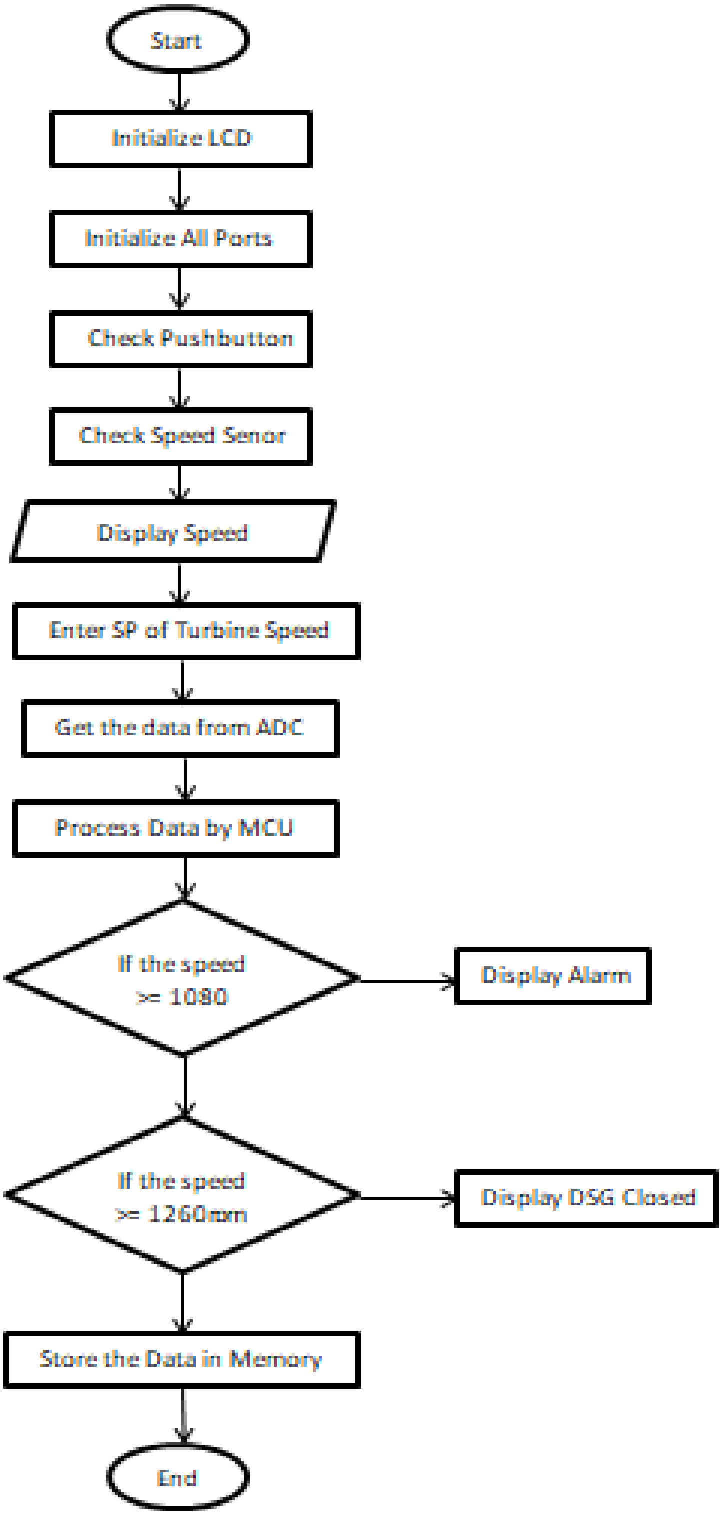


Fig. 3.18 General Flowchart for the system

Speed Reading Algorithm:

The Arduino Uno has 14 digital inputs output pins; it also has high and low voltage references which are always in our code set to 5V and 0V, respectively. Also the MCU has 6 analog inputs. The speed sensor analog voltage output is applied to any one from 6 Analog in channel hence the algorithm to read speed will be:

(*i*) Initialize the microcontroller:

- ADC configuration: ADC channels are voltage references are 0 V and 5 V.
- Clock = 16 MHz

(*ii*) Set ADC channel for analog, Read ADC value.

(*iii*) Calibrate the obtained digital value to actual speed reading using :

Count * 60 = speed (rpm)

Then the flow chart of the algorithm will be as in Fig. 3.19.

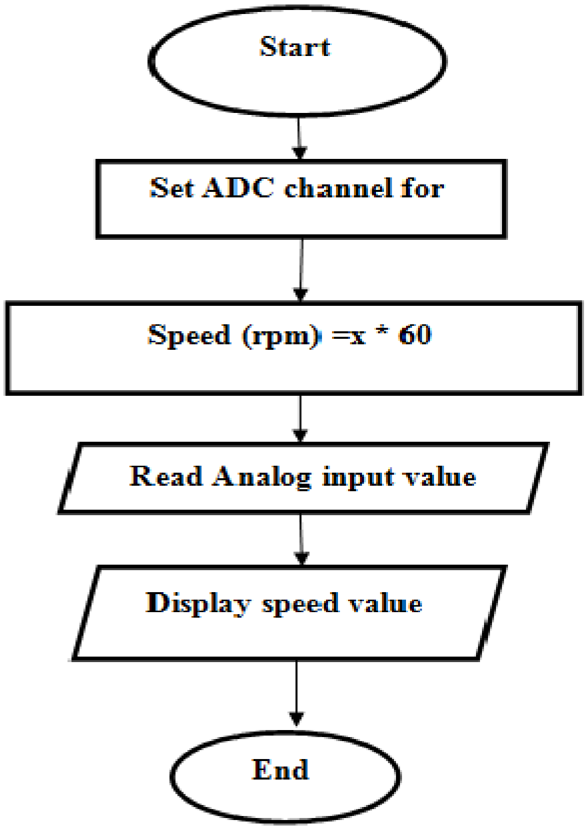


Fig. 3.19 Speed Reading Algorithm

3.4.4 Basics of Programming Languages

All sequential programming languages have four categories of instructions.

First are operation commands that evaluate an expression, perform arithmetic, toggle states of I/O lines, and many other operations.

Second are jump commands that cause the program to jump immediately to another part of the program that is tagged with a label. Jumps are one way to break out of the normal line-by-line processing mode. For example, if you want a program to repeat over and over without stopping, have the last line of the program be a jump command that takes the program back to its first line.

Third are branch commands that evaluate a condition and jump if the condition is true. For example, you might want to jump only if a number is greater than zero. Or, you might want to jump only if the state of an I/O line is low.

Fourth are loop commands that repeat a section of code a specified number of times. For example, with a loop you can have a light flash on and off exactly six times.

Most programming languages contain a relatively small number of commands. The complexity of computers comes from combining and repeating the instructions several million times a second.

Arduino Programming Language

The Arduino runs a simplified version of the C programming language, with some extensions for accessing the hardware.

All Arduino instructions are one line. The board can hold a program hundreds of lines long.

The Arduino executes programs at about 300,000 source code lines per sec.

Creating a Program:

Programs are created in the Arduino development environment and then downloaded to the Arduino board. Code must be entered in the proper syntax which means using valid command names and a valid grammar for each code line. The compiler will catch and flag syntax errors before download. Sometimes the error message can be cryptic and you have to do a bit of hunting because the actual error occurred before what was flagged.

Although your program may pass cleanly through the syntax checker, it still might not do what you wanted it to. Here is where you have to hone your skills at code debugging. The Arduino execute the task what you told it to do rather than what you wanted it to do. The best way to catch these errors is to read the code line by line in the computer. Having another person go through your code also helps. Skilled debugging takes practice.

Program Structure:

All Arduino programs have two functions, setup and loop. The instructions place in the start up function is executed once when the program begins and is used to initialize. Use it to set directions of pins or to initialize variables.

3.5 Proteus ISIS 7

Proteus 7.0 is a Virtual System Modelling (VSM) that combines circuit simulation, animated components and microprocessor models to co-simulate the complete microcontroller based designs. This is the perfect tool for engineers to test their microcontroller designs before constructing a physical Prototype in real time. This program allows users to interact with the design using on-screen indicators and/or LED and LCD displays and, if attached to the PC, switches and buttons. One of the main components of Proteus 7.0 is the Circuit Simulation. Proteus VSM comes with extensive debugging features, including breakpoints, single stepping and variable display for a neat design prior to hardware prototype.

In summary, Proteus 7.0 is the program to use when you want to simulate the interaction between software running on a microcontroller and any analog or digital electronic device connected to it .

ISIS is a software program that enables users to design electronic circuits schemes or printed circuit boards. The main advantage offered by this program to others such as or CAD is real –time simulation of all types of electronic circuits, watching clear graphics defaults sets… for interactive simulation of circuits, it was implemented PROTEUS VSM module. Thus using this module you can draw a complete circuit for a microcontroller based system, and then it can be tested interactively. ISIS also provides to the user option for customizing parts and components involved in making a circuit. ISIS uses the following file types and formats: design file (DSN), backup files (DBN), Section- Files (CES), Module Files (MOD), Library Files (LIB) and Net list-Files (SDF). Type design files contain all the information about a circuit. Backup of type design files are created when the saving is carried over an existing file. Section files can be exported and red in another drawing and those with net list type are produced through exporting in Pros Spice and ARES. Creating a new design is made by New Order Design. Start up of this command removes all existing design data and displaying a blank standard interfaces A4 size. The created a design can be done in three modes. In DOS command line, by ISIS command <design name> or selecting Open Design once the program is on, or double – clicking the file in Windows Explorer.

3.6 Study and analysis the hydraulic system of DSG:

Hydraulic systems are a power transmission chains which convert mechanical energy into pressure and flow and back into mechanical movement again.

Generally the initial mechanical energy is rotational movement which is created with an internal combustion engine or an electric motor. The transmission of pressure and flow is done with hydraulic oil and the final movement can be either rotational or linear movement.

The hydraulic system of the DSG for the the Pump Turbine No. 2 is shown in Fig. 3.20

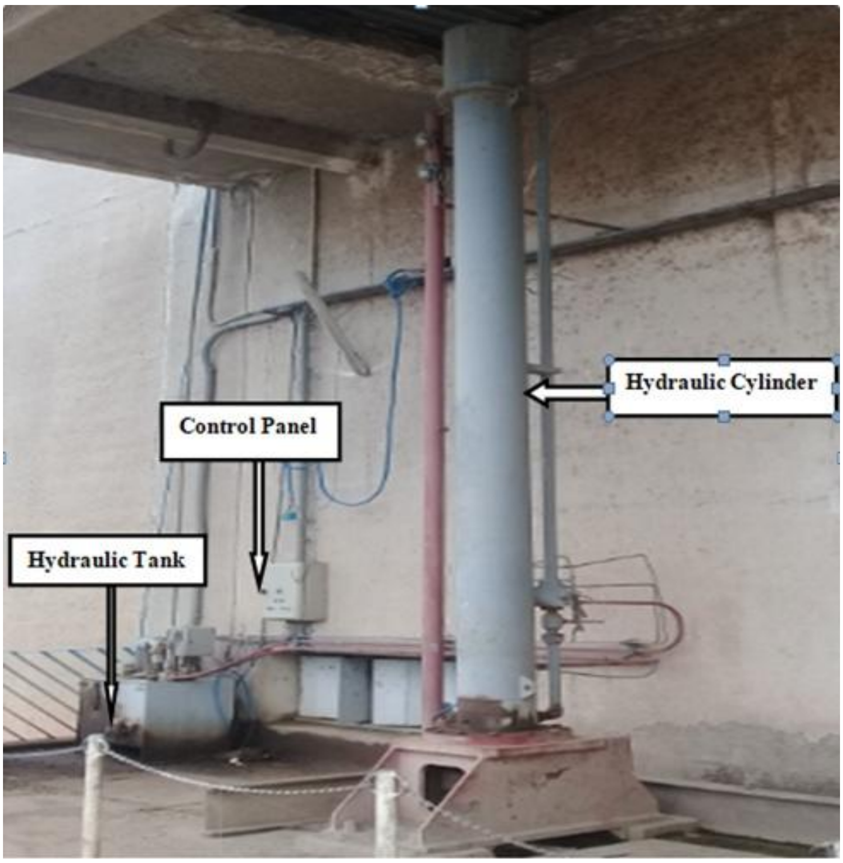


Fig. 3.20 Hydraulic System of the DSG for the Pump Turbine

The signal that sent from Arduino Uno MCU used to open valve to close the DSG of the pump turbine by gravity.

The technical data of the hydraulic system for the DSG of the pump turbine is shown in Table 3.6.

Table 3.6 Technical data of the DSG hydraulic system

Cylinder diameter	280, mm
Rod diameter	120, mm
Stroke	4800, mm
Nominal pressure	160, bar
Test pressure	240, bar
Design force	800, kN
Electrical motor power	7.5, kW
Pump delivery	22, l/min
Tank capacity	300, liters

To avoid water hammer due to close the DSG the closing time of the DSG must be as follow:

$T_c > 2l/C$

Where:

T_c = closing time, s

l = pipe length, m

C = sound speed in water, m/s

$$C = \sqrt{\frac{K}{\rho}} \qquad (3.1)$$

Where:

K = bulk modulus of elasticity (N/m^2) = 2.2×10^9 N/m^2

ρ = density (kg/m^3) = 10^3 kg/m^3

Using equation (3.1) yields the result.

C = 1483 m/s

The length of path from intake gate to DSG = 24 m. So

$$(2l/c)_{DSG} = 0.032 \text{ sec}$$

To avoid water hammer the DSG closing time must be more than 0.032 sec.

The flow rate through flow control valve (FCV) is governed by:

$$Q_{FCV} = C_V\sqrt{\frac{\Delta P}{SG}} \quad \ldots\ldots\ldots\ldots \quad (3.2)$$

Where:

Q_{FCV} = flow rate, m^3/h

C_V = valve sizing coefficient of FCV

ΔP = pressure change, bar

$$\Delta P = P_{in} - P_{atm}$$

$$P_{in} = F_{load}/(A_p - A_r)$$

$$A_p = \pi /4 * Dp^2$$

$$A_r = \pi /4 * D_r^2$$

Where:

F_{load} = Design force, N = 800 kN

D_p = Piston diameter, m = 0.28 m

D_r = Rod diameter, m = 0 .12 m

A_p = effective piston area, m^2 = 0.0616 m^2

A_r = rod area, m^2 = 0.0113 m^2

SG = specific gravity of iso VG 32 oil, m^3/kg = 0.8618 m^3/kg

P_{atm} = Atmospheric pressure, bar = 1.013 bar

By applied equation (3.2) the result is:

Q_{FCV} = 0.23 m^3/s and this is max flow through flow control valve since Valve coefficient =1

But the $Q_{ext} = Q_{FCV}$

$$Q_{ext} = (A_p - A_r) \times \frac{L}{T_c} \ldots\ldots\ldots\ldots (3.3)$$

Where:

Q_{ext} = flow rate from cylinder in extending, m^3/s

L = Cylinder stroke, m = 4.8 m

By applied equation (3) yields for this relation:

$$Q_{ext} = \frac{0.24}{T_c}$$

Since the minimum closing time for the DSG of the pump turbine, T_c equal 60 second. The FCV opening can be adjusted to select the suitable time and speed for closing the DSG.

The DSG Fig. 3.21 is closed by gravity without intervention of the pump unit. The solenoid valve is energize, then the two way valves opened and the pressurized oil present in the lower chamber of the cylinder due to the weight of DSG flows from the lower chamber into the upper chamber through flow regulation valves, which control the closing speed Fig. 3.22.

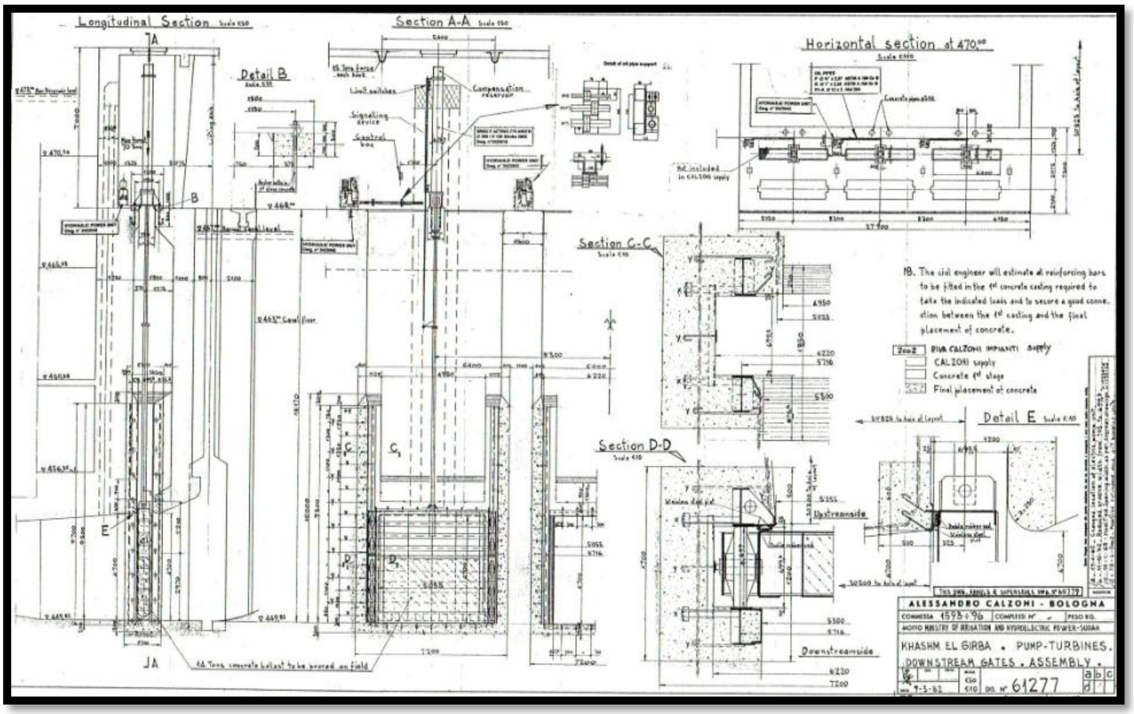


Fig. 3.21 Sectional view of DSG for the pump turbine

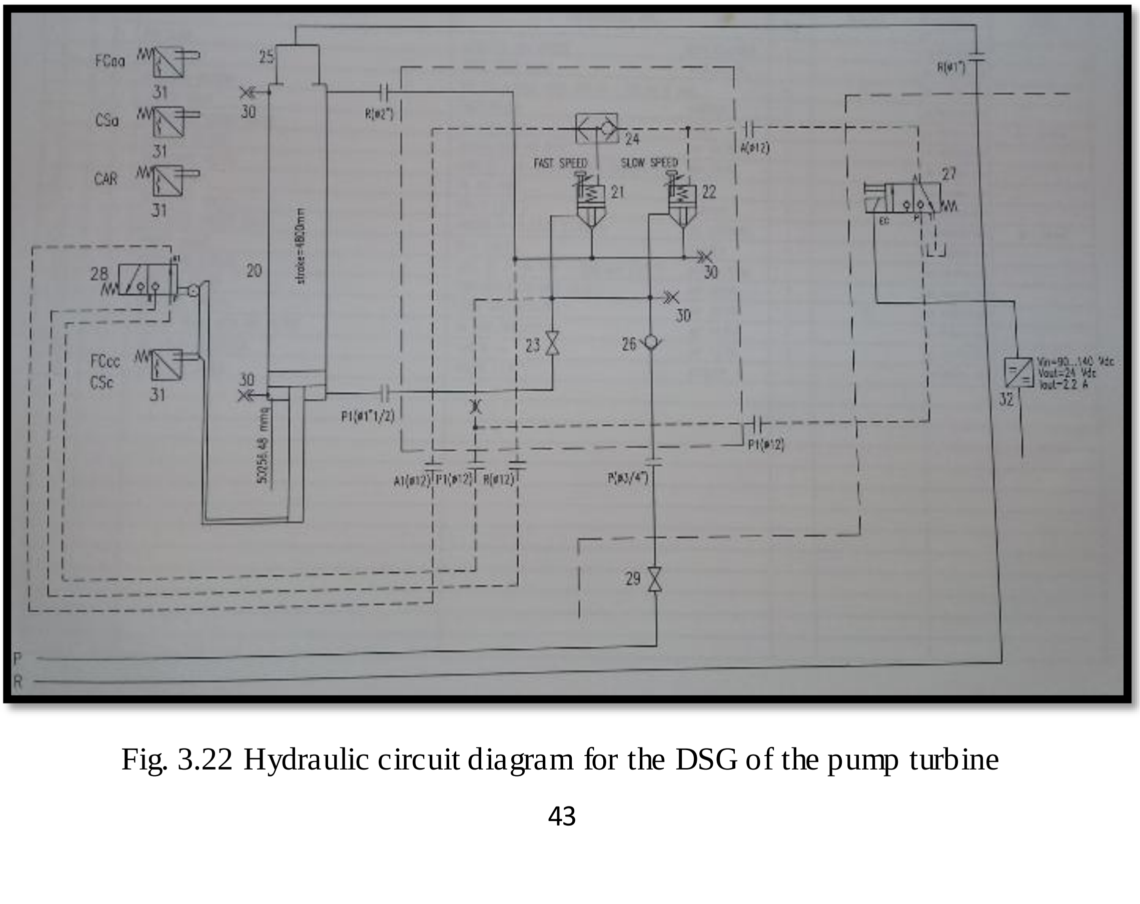


Fig. 3.22 Hydraulic circuit diagram for the DSG of the pump turbine

CHAPTER IV

RESULTS AND DISCUSSIONS

CHAPTER IV

RESULTS AND DISCUSSIONS

4.1 Results:

The simulation results of complete circuit carried out using Proteus software. The results obtained from the Arduino Uno interfaced with IR sensor, DC motor, pushbutton, LCD, buzzer and LED's. The LCD is used to monitor the sensor readings and show the condition of the system (overspeed and trip) Fig. 4.1. The LEDs are also connected for identification of the sensor working properly.

The performance of the system depends on working of each individual device. Hence overall performance of the unit has been checked and has been satisfactory.

Fig. 4.1 Microcontroller Based Speed Controller

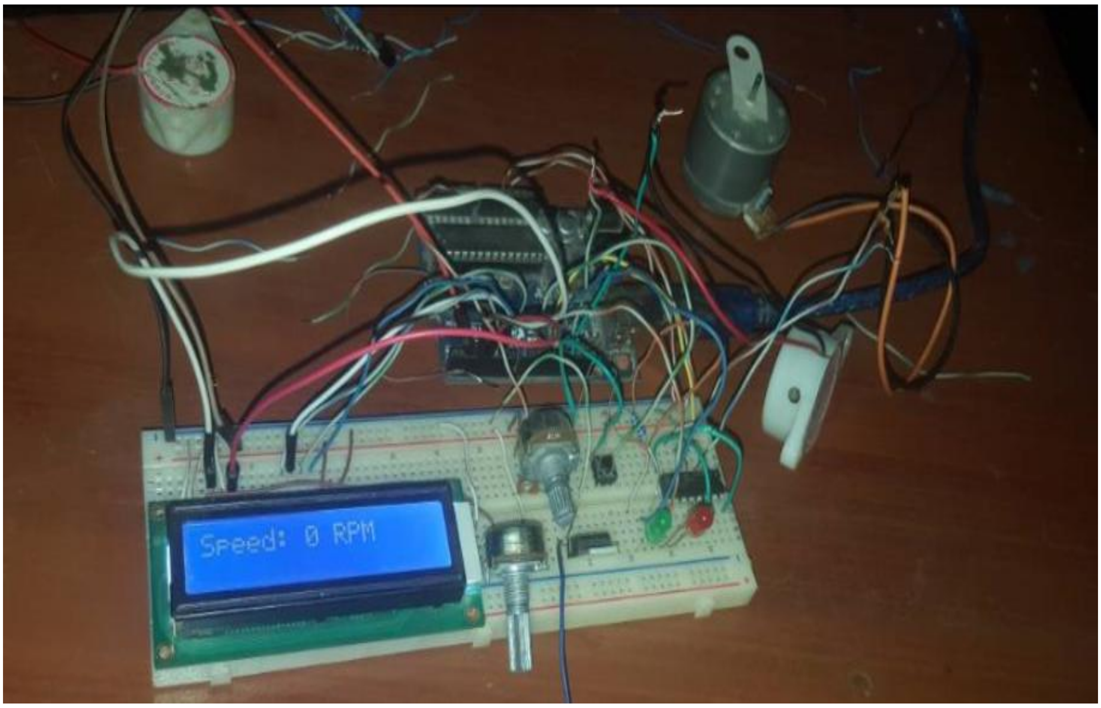


Fig. 4.2 Model of DSG control system

The model is shown in Fig. 4.2 was running for testing in laboratory kids after complete. When the model is operate the DC motor is running and the speed of the motor is reading by IR sensor, the reading showing on the LCD when speed of motor reach to 1080 rpm the overspeed text appear on the LCD and red LED light and buzzer sound activated.

When the speed reach to 1260 rpm the trip text appear on the LCD and the green LED light.

4.2 Discussions

The speed of the unit normally constant at specific speed when the unit trip due to any cause while the unit loading the speed increase. For maintaining the speed not exceed and reach to run way speed a control system is required. An Arduino Uno board based microcontroller is applied in contrast to the control DSG of the unit which is built on the electronic board. The opportunity exists to optimize the control loop to achieve the best possible accuracy in each case.

When the model of the DSG Control system tested in laboratory kids the microcontroller sent signal to the red LED to light and to buzzer to do sound and the overspeed expression wrote on LCD when speed of motor reach to 1080rpm which represent the overspeed level one and equivalent to 144% from normal speed of the DC motor. And when the speed of motor increased and reached to 1260rpm which represent overspeed level two and equivalent to 168% from normal speed the microcontroller sent signal to light green LED and this signal represent signal that sent to close the DSG of the pump turbine in order to stop the unit and the trip text appear on the LCD.

Also when pushbutton pressed the microcontroller sent signal to the light green LED and as remembered above this signal that light the green LED which represent the signal that result to close the DSG of the unit and the speed of motor go to zero. And TRIP text wrote on the LCD.

CHAPTER V

CONCLUSION AND RECOMMENDATION

CHAPTER V

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

The research presents a design of control DSG of the PT2 in Khashm Algirba power station. To achieve this system was developed with compatible software in MikroC (MikroC is an Embedded-C compiler that allows you to convert a code written in C language to machine language), so this system can provide optimum overspeed protection by the control technique which has been designed.

The result showed the simulation and model for the system, since when the speed of the unit reach to 1260 rpm the Arduino Uno MCU sent signal to the DSG valve to close the DSG of the pump turbine and prevent unit from run way speed.

According to study and analysis the hydraulic system of the DSG find the minimum time to closing the DSG is 72 sec.

According to the study and analysis of various parts of the system, a design has been carried out. The results obtained from the measurement in laboratory kids shown that the system perform well.

5.2 Future scope

The performance of microcontroller based overspeed protection has been found on the expected lines. However, there exists a scope for further improvement in its speed and PC interface software for post data analysis.

Adding the control system of the DSG of the PT2 to the basic control system of the unit.

References:

Atmel Corporation (2016). 8-bit AVR Microcontrollers Atmel Corporation. Atmel-42735B-ATmega328/P.

Chirag Trivedi, Michel J. Cervantes and B. K. Gandhi (2016). Investigation of a High Head Francis Turbine at Runaway Operating Conditions Academic Editor: Juan Ignacio Pérez-Díaz,.

Christophe Nicholet, Sebastien ALLIGNE, Basile Kaw Kabani, Jiri Koutnik, Jean, Jacques Simond and francios Avellan (2009). Stability study of Frances pump turbine at run way- IHAR International meeting of the work group on cavitation and dynamic problems in hydraulic machinery and systems, , Brno, Czech Republic.

Ewetumo Theophilus (2012). Development of microcontroller base automatic level of discomfort Index Measurement using Thom Model. ARPN Journal of Science and Technology VOL. 2, NO. 3, April 2012 ISSN 2225-7217,

Gerald Felix1and Jiang De Ning. (2015): Enhanced Bank Vault (Strong Room) Security System Design. International Journal of Science and Research (IJSR).

Helena Ramos and A. Betamio De Almedida (1996). Modelling and practical analysis of the transient overspeed effect of small Francis turbines.

H Hosseinimanesh, T. C. Vu, C. Devals3, B. Nennemann and F. Guibault (2014). A steady-state simulation methodology for predicting runaway speed in Francis turbines. IOP Conf. Ser.: Earth Environ. Sci. 22 032027.

Kamaran Jamal Muhedeen (2011). Hydroelectric power generation. Licentiate electrical engineer.

http://keu92.org/uploads/Search%20engineering/HYDROELECTRIC%20POWER.pdf

K. D. Young and H. G. Kwatny (1981). An application of a variable structure servomechanism design to turbine overspeed protection control. pp.3045-3052.

Qinfen Katherine Zhang numerical Modelling of Active Hydraulic Devices and Their Significance for System Performance and Transient Protection.

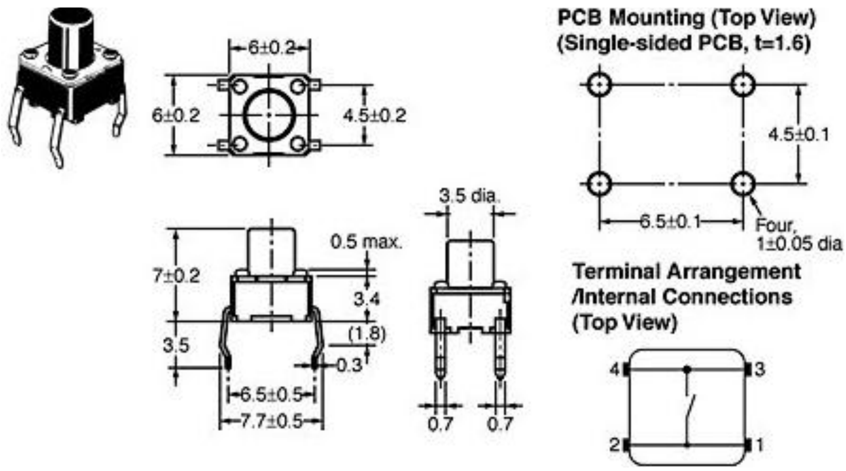
Shwetha K.S. Shwetha K.S., Snehith Osman, Mohammed Shahid (2015). A Study on Developments in Wireless Visible Light Communication: *IJCSMC, Vol. 4, Issue. 2, pg.325 – 330*

Varuninder Singh (2006). Design and Development of Micro controller Based Temperature and Humidity Controller for Infant Incubator : Regd No: 8044218

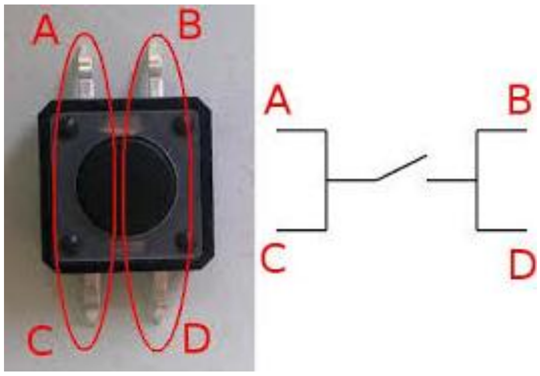
Voith Turbo Gmbh & Co. KG (2012).

http://pdf.directindustry.com/pdf/voith-turbo/overspeed-protection-turbines-cto/39449-431673.html

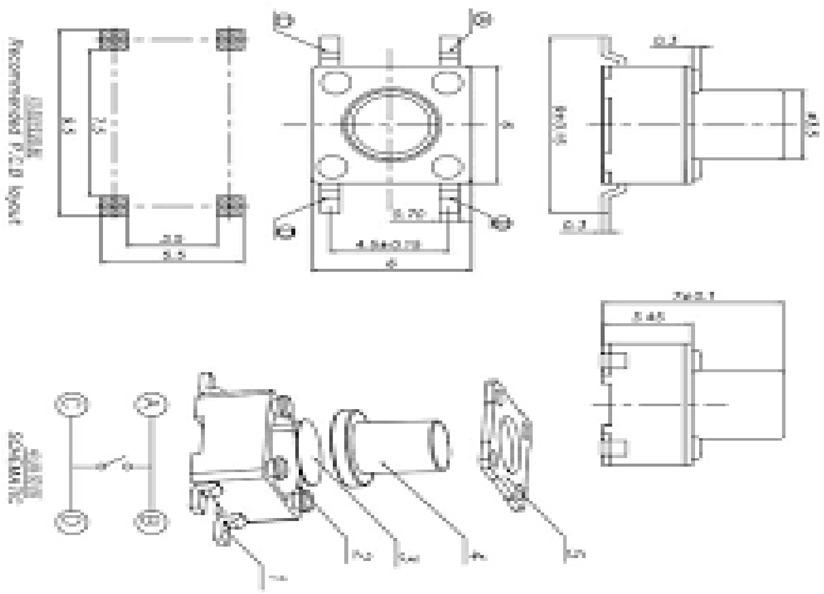
Appendix A



Appendix (A-1) Pushbutton Schematic



Appendix (A-2) symbol of pushbutton



Appendix (A-3) Pushbutton assembly

Appendix B

Code

```
// motor speed//
#include <TimerOne.h<
#include <LiquidCrystal.h<
unsigned int counter = 0;
LiquidCrystal lcd (3, 4, 5, 6, 7, 8);
void docount()  // counts from the speed sensor
{
  counter++;  // increase +1 the counter value
}
void timerIsr()
{
  Timer1.detachInterrupt();  //Stop the timer
  Serial.print("Speed'');
  int rotation = (counter * 60); //rotation perminuite
  Serial.print(rotation, DEC);
  counter = 0; //  reset counter to zero
  Timer1.attachInterrupt( timerIsr );  //enable the timer
  lcd.setCursor(0, 0);
  delay(4000);
  lcd.clear;()
  lcd.setCursor(0, 0);
  lcd.print("Speed'');
  lcd.setCursor(7, 0);
  lcd.print(rotation, DEC);
```

```
  lcd.print(" RPM’’);
{
//motor pins
int Motor11 = 11;
int potdeger = A0;
int deger;
void setup} ()
  Timer1.initialize(1000000); // set timer for 1sec
  attachInterrupt(0, docount, RISING);  // increase counter when speed sensor pin
goes High
  Timer1.attachInterrupt( timerIsr ); // enable the timer
  Serial.begin (9600);
  lcd.begin(16, 2);  pinMode(12, OUTPUT);
  pinMode (Motor11, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(9, INPUT);
  pinMode(13, OUTPUT);
  digitalWrite(Motor11, HIGH);
{
void loop} ()
  int Stop = digitalRead(9);
  deger = analogRead(potdeger);
  deger = map(deger, 0, 1023, 0, 225);
  analogWrite(Motor11, deger);
  if (counter * 60 >= 1080&&counter *60<=1260)
}
    digitalWrite(10, HIGH);
```

```
    delay(500);
    digitalWrite(10, LOW);
    delay(500);
    lcd.setCursor(0, 1);
    lcd.print(" OVER SPEED);
{
  else if (counter*60>1260)
}
    digitalWrite(10,LOW);
    digitalWrite(12,HIGH);
    digitalWrite(13,HIGH);
    delay(2000);
    digitalWrite(12,LOW);
    digitalWrite(13,LOW);
  lcd.setCursor(0, 1);
    lcd.print(" TRIP’’ );
    digitalWrite(11,LOW);
    delay(10000);
{
    else
{ digitalWrite(10,LOW);
    digitalWrite(12,LOW);
    digitalWrite(13,LOW);
  if (Stop == HIGH)
  { digitalWrite(12,HIGH);
  delay(2000);
  digitalWrite(12,LOW);
```

```
    digitalWrite(11, LOW);
    delay(10000);
}
}}
```