Microcontroller based Numerical Relay for Induction Motor Protection

Vikramsinh N. Kathad[#1], Rashesh P. Mehta[#2], Ankit Shah[*3]
#Department of Electrical Engineering, B.V.M.Engineering College, Sardar Patel University
Vallabh Vidyanagar, Dist.-Anand, Pin-388 120, State-Gujarat, India.
kathadvikram@yahoo.co.in,
rpmehta@ieee.org
*B.B. Institute of Technology
Vallabh Vidyanagar, Dist.-Anand, Pin-388 120, State-Gujarat, India.
ankit_eng@indiatimes.com

***Abstract*— This paper presents the protection arrangements for small and large 3-phase induction motors either of squirrel cage or wound rotor type. The design and operation algorithm for the protection of Induction Motor is done using MSP430F148 microcontroller. Various fault conditions for the protection of Induction Motor relay are realized by calculating the values of V and I during the fault condition. The signals are taken from the supply line converted to digital signals which then are fed to microcontroller having a program written in it to estimate the values of V and I. The calculated values of V and I are compared with set values to decide about the action to be taken. Design and development of microcontroller based induction motor protection relay is recent trend in field practice. One such design effort has been made by authors.**

***Index Terms*— fault, induction motor protection, microcontroller, numerical relay, simulation**

I. INTRODUCTION

There are many types of induction motors used in practice. Small motors having b.h.p. of few tens of h.p. and rated for 415 volts can be protected by starters of various kinds having in built thermal overload relay facility and so often protection of short-circuits. Large three phase induction motors (ranging from 100 h.p. to 5000 h.p. or more) and in medium voltage range (of the order of 3.3 kV, 6.6 kV, 11 kV etc.) are used for running power station auxiliaries and in large industries. These motors are controlled by circuit breakers and associated protective relays. Such motors need comprehensive protective arrangement.

Power utilities traditionally use electromechanical and solid state relays for protecting Induction motors. With advance in technology, protective relays have progressed from electromechanical, to solid state and to microprocessor based relays. The increased growth of power systems both in size and complexity has brought about the need for fast and reliable relays to protect major quipment and to maintain system stability. With the development of economical, powerful and sophisticated microprocessors, there is a growing interest in developing microprocessor based protective relays, which are more flexible because of being programmable and are superior to conventional electromagnetic and static relays. The main features which have encouraged the design and development of microprocessor based protective relays are their economy, compactness, reliability, flexibility and improved performance over conventional relays.

A. Faults in Induction Motor:

The probable faults in Induction motor can be classified into two broad categories:

Stator Faults

Rotor Faults

Short circuit in the stator winding is harmful because it will involve high stator currents and unsymmetrical faults will cause unbalance and hence negative sequence currents are generated and consequently rotor is heated. Rotor faults cannot be normally developed in squirrel cage induction motors. However, such faults are possible in wound rotor motors. The consequence of such fault is to increase stator current. Moreover, vibrations are caused in the rotor which may endanger the motor.

B. Abnormalities in Induction Motors:

Following abnormal conditions are known to exist for an induction motor.

1. Overloading
2. Single Phasing

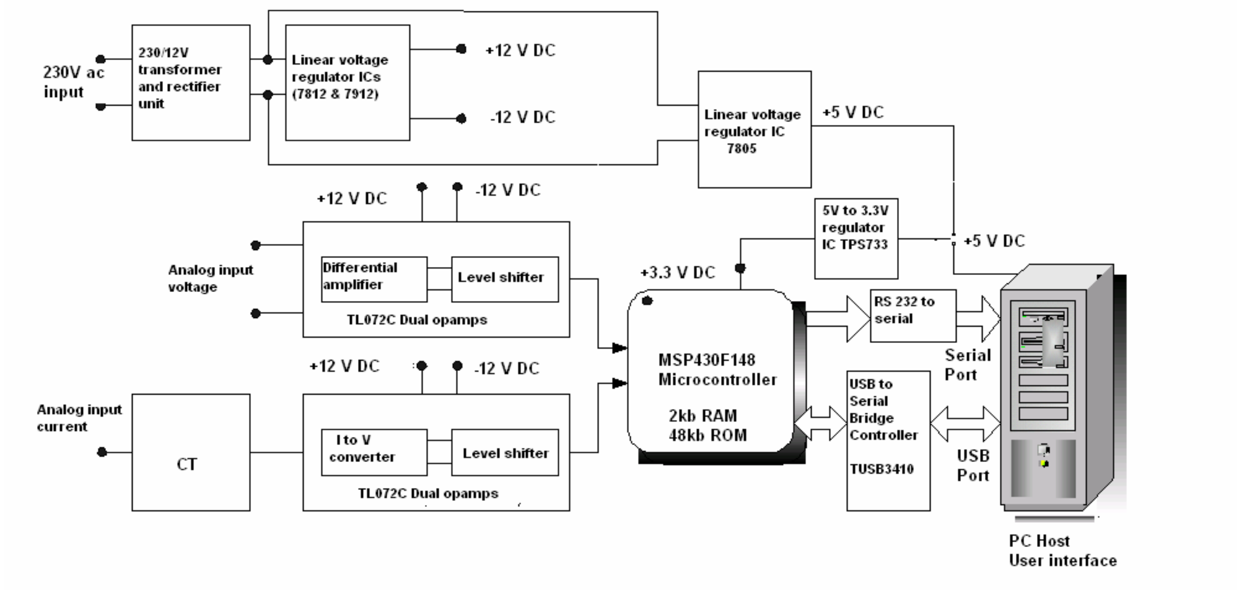

Fig. 1 Block diagram of the project

3. Voltage Unbalance
4. Reverse phase sequence
5. Under voltage
6. Stalling

These abnormalities can be cleared by using the Numerical Induction Motor Relay developed in our project. This relay is designed with the use of Microcontroller MSP430F148 of Texas Instruments Co. Ltd. Hardware circuit is designed with the help of this controller which we have used for the development of the relay.

The microprocessor-based Induction motor relays like electromagnetic or static relay sense the fault from the power system data such as voltage, current, phase angle, and frequency.

A typical numerical relay consists of input from CT & PT, analog low pass filter, I to V converter, voltage step down circuit, sample & hold circuit, multiplexer, ADC and microprocessor.

The commonly used components of numerical relays are as follows:

1. Combinational circuits.
2. Sequential circuits.
3. Filters (Analog and Digital).
4. Sample & Hold circuits.
5. Reactive circuits.
6. Multiplexers.
7. ADC & DAC.
8. Microprocessors.

The current and voltage for microprocessor-based Induction motor relays are measured by conventional CT & PT transducers and compared with preset quantities to decide the fault. The relay receives the signal from supply line through conventional CT & PT transducers, surge filters, multiplexer, sample & hold and ADC. These signals are current and voltage. Microprocessor processes these signals and decides whether there is fault or not. Accordingly relay gives signal to circuit breaker to trip. Microprocessor has serial and parallel port interfacing for communication of relay with PC.

II. RELAY HARDWARE

The block diagram of “Numerical Induction Motor Relay” using microcontroller MSP430F148 is shown in Fig. 1.

The voltage signal is given to differential op-amp for stepping down the voltage signal to desired level as ADC does not allow voltage signal magnitude greater than 3.3Volt. Then this step down voltage signal is shifted with the help of level shifter circuit because ADC inhibits bipolar signals (positive as well as negative).

In order to get low voltage up to 3.3Volt and unipolar input, we have used voltage step down circuit along with level shifter using op-amps (TL072C).

Then this shifted signal is passed through active low pass filter as anti aliasing filter to avoid aliasing errors in a sample data.

The current to voltage converter is used to convert current signal in to the voltage signal which is then fed to level shifter circuit to get unipolar signal. To reduce the hardware circuitry we have used MSP430F168 microcontroller, which has inbuilt ADC with 8 channel multiplexer and sample & hold circuit. Voltage and current signals after conditioning are fed to MSP430F148. Multiplexer is used to take one signal at a time, and then this signal will be converted to digital signal by using ADC. The microcontroller receives the conditioned signals and starts computing V and I.

Different signal conditioning circuits require +12 volt and -12 volt DC supply. For this purpose, rectifier circuit with linear regulator ICs 7812 & 7912 is used. 230 volt / 12 volt transformer is used to step down the voltage. Communication between microcontroller and PC can be done through serial communication port, but we have used serial communication to USB converter IC TUSB3410.

This IC converts serial data to USB compatible data so that we can directly connect our relay to the USB port of the PC. Main advantage of doing this is that it can be even directly connected to laptops, which do not have any serial communication port. In case of PC, serial communication can also be used, for which IC RS232 will be required instead of TUSB3410. MSP430F168 requires 3.3V DC for its operation. This voltage is made available using IC TPS733, which converts 5 volt D.C. available from USB to 3.3 volt D.C.

MSP430F168 Architecture:

MSP430 incorporates a 16-bit RISC CPU, peripherals, and a flexible clock system that interconnect using a von-Neumann common memory address bus (MAB) and memory data bus (MDB).

Key features of the MSP430x1xx family include:

- Ultralow-power architecture extends battery life.
- 0.1-µA RAM retention.
- 0.8-µA real-time clock mode.
- 250-µA / MIPS active.
- High-performance analog ADC - ideal for precision measurement.
- 12-bit or 10-bit ADC — 200 ksps
- 12-bit dual-DAC
- Comparator-gated timers for measuring resistive elements
- Supply voltage supervisor
- 16-bit RISC CPU enables new applications at a fraction of the code size.
- Large register file eliminates working file bottleneck
- Compact core design reduces power consumption and cost
- Optimized for modern high-level programming
- Only 27 core instructions and seven addressing modes.
- Extensive vectored-interrupt capability.

III. DEVELOPMENT OF APPLICATION PROGRAM

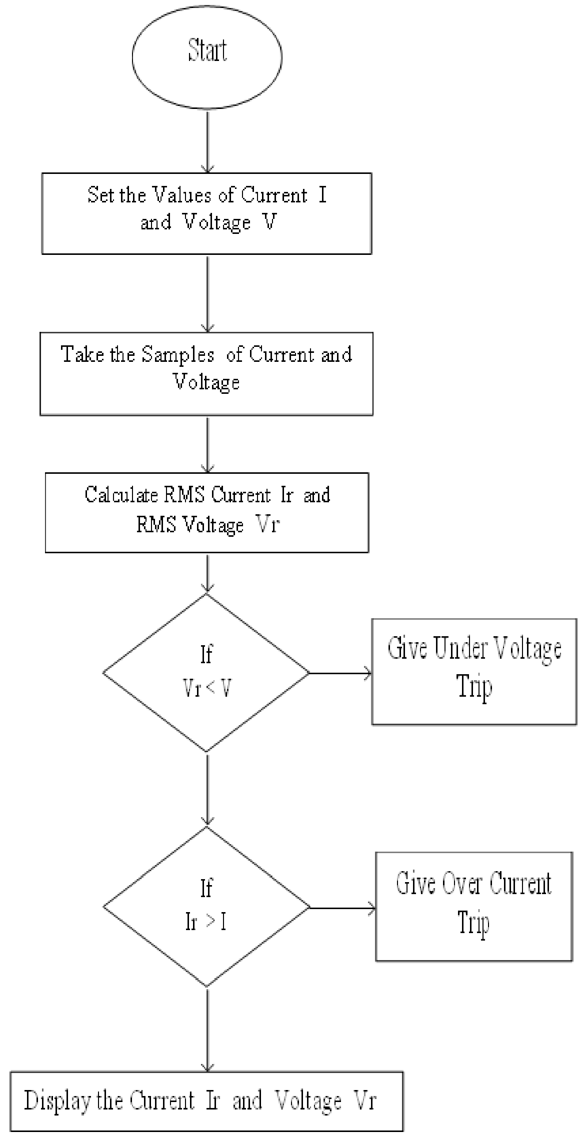

Fig. 3 Flowchart of the program

A. *Programming Tools:*

1. Programming via JTAG Interface: JTAG Interface is payable loader which cost about thousand of rupees. After buying JTAG, program can be loaded through windows operating system. But in our application we have loaded our program through BSL which is free software available on internet. But it needs Linux as an operating system.
2. Programming via Bootstrap Loader (BSL) interface.
3. BSL is a factory-masked loader program to communicate via serial link with embedded memory in the microcontroller. The flow chart shows the flow of the program. First we will set the values of the reference parameters currents and voltage. Now as we start the Induction Motor controller will start taking

samples of the current and voltage. By doing mathematical calculations it will calculate RMS Values of Current and Voltage. Now as we have set values and instantaneous values of current and voltage we can compare them with the help of comparator in the micro-controller and can take the decision for the tripping of the Induction Motor.

4. MSPGCC tool chain: It is a part of GNU C compiler, an open source environment having GNU C Compiler, linker, assembler and supporting utilities developed for MSP430 family of Microcontrollers.
5. Driver Program / Interfacing Program :

It is the program developed in C for connecting PC to microcontroller. With the help of GCC, C program is converted into assembly program and hex which is then loaded in our microcontroller MSP430F148 by BSL (Bootstrap Loader) interface. The flowchart is shown in figure 3.

IV. Results

By using this developed Numerical Induction Motor protection relay we have developed several protections for the simulation purpose. The panel connections are shown in figure 4.

We can do the protection of the 3-phase Induction Motor from several faults as explained earlier. We have simulated Over Current, Under Voltage and Start-up time monitoring for the 3-phase Induction Motor.

Results obtained during the performance are as under. For the Over Current and under voltage set values are as shown here: I = 1.20 A, V = 200 V, Start up time = 30 seconds.

TABLE I
Results of testing

Sr. No.	Voltage Vr	Current Ir	Trip signal	Display
1.	235.8V	1.08A	No Tripping	I and V will be displayed
2.	258.0V	1.26A	Trip	Over Current
3.	180V	0.81A	Trip	Under Voltage

With the above set values by performance we have obtained the results as shown below in Table 1.

Fig. 4 Panel connections of the numerical relay

By performing above operations, particular tripping signals are generated and displayed on the LCD Display.

V. CONCLUSION

At the end we have successfully designed Numerical Induction motor Relay by using micro-controller MSP430F148 which can be used for the complete protection of the Induction motors. By using this relay we have simulated several protections in my laboratory also. It is giving continuous monitoring of the line current, voltage and frequency. At the same time it is displaying the same.

The compatibility, economical viability and functional flexibility of this Numerical Induction motor relay make it advantageous over conventional electromagnetic and static relays.

VI. ACKNOWLEDGEMENT

We express our gratitude to B.V.M Engineering College for the grant and support provided for my project. We are grateful to Prof. Mukul Chandorkar & project staff Miss Sneha Mistry of Department of Electrical Engineering, IIT, Bombay for providing support .

VII. References

[1] Date, Oza, Nair *Power System Protection*, Bharti Prakashan, 2002.
[2] B.Ram, *Fundamentals of Microprocessors and Microcontrollers*, S.Chand & Co., 2002
[3] Chris Nagy , *Embedded systems design using the TI MSP430 series*, Newnes Press, 2003
[4] *MSP430X1XXX Family User's Guide*. Texas Instruments, 2005