RESEARCH ARTICLE OPEN ACCESS

Design and Implementation of Microcontroller Based Self-Switching Control and Protection System for Twin Pumps

Hrudaynath Yelgudkar*, Satyajeet Waghmare*, Prasanna Parmar*
*(BE, Department of Electronics and Telecommunications, Pune Vidyarthi Griha's College of Engineering and Technology, Savitribai Phule Pune University, Pune.

ABSTRACT
This paper apprises reader with a unique, modular and comprehensive control system designed using embedded systems approach (microcontroller based), primarily for efficient use of twin (two or couple of) pumps working simultaneously in a given environment while constantly monitoring critical parameters such as current, voltage, temperature, water level in reservoir etc. for protection purposes. Though this control system is developed for pumps, it can be seamlessly adapted for controlling similar loads. We have developed the system using ATMmega-32 microcontroller of AVR family. The critical parameters were monitored using ADC port (analog-to-digital-converter port) of the microcontroller. Motivation behind developing this system was to replace classical 'dedicated integrated circuit' based control system with more intelligent, compact, programmable and upgradable system besides lowering its cost aspects, using 'embedded systems'.

Keywords: AVR, ADC, Embedded Systems, Microcontroller, Twin Pumps

I. INTRODUCTION

Today, it is imperative to have a credible, robust and an all-inclusive control system for electrical loads in industry applications or domestic use. Therefore developing a control system having a lucid user interface, fulfilling all requirements for efficient, safe working of loads and good communication with outer world besides having a lower power consumption and smaller cost-space footprint, would be of a great relevance. The aim of this project is to develop a control system, which would match with the ideals mentioned above to a reasonable extent.

The places such as industries, dams or large residential complexes often require more than just a single pump, typically two pumps (motors), to draw the water out from the reservoir tank to the storage tank. Pumps (motors), just like any other electro-mechanical equipment, can't be built to work forever. That is, they have a limited life-time, for which they can run flawlessly.

However, this flawless running time can be extended with better and smart utilization of pumps as well as by ensuring their protection from faults such as over-voltage or over-current. The system that this paper talks about has been designed and implemented for twin pumps system to ensure these two purposes, better utilization and protection of pumps respectively, so that the flawless working of pumps can be pushed well beyond its usual life time.

The existing systems used serve the same purposes have a complex interconnection of dedicated ICs which not only makes the system bulky, but also costly, one time programmable and very difficult to upgrade.

With the use of microcontroller, we were able to overcome the fallacies in dedicated IC based system and also provide the lucid menu-driven user interface to operate and navigate through the various functionalities of the user interface (UI) via LCD and keypad interface.

II. SYSTEM ARCHITECTURE

Shown below is the block diagram of the implemented control system:

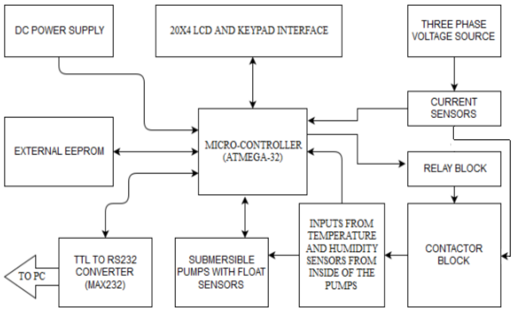


Fig. 2.1 System Block Diagram

2.1 Microcontroller

As shown in **Fig. 2.1**, ATmega-32 microcontroller [1] is central to every other block. It acts as a brain of the system. The '.hex' file of the source code developed in 'Embedded C' programming language has been downloaded on the microcontroller, so as to dictate the working. The source code takes care of the logic of the system's functionality and hence it substitutes the majority of the intricate hardware of existing application-specific-integrated-circuit (ASIC) based systems, hence greatly reducing the cost and the size.

ATmega-32 is an 8-bit microcontroller of Atmel's AVR family. The source code for the same has been developed and compiled in Atmel Studio 6.2 IDE. Software named ProgISP has been used to download the compiled '.hex' file of the source code on the microcontroller.

ATmega-32 comes with a 10-bit, 8 channel on-chip analog-to-digital-converter module (ADC). [1] Hence, interfacing of various sensors to monitor critical parameters has been facilitated without having to use an external ADC, thus making it further compact.

2.2 LCD- Keypad Interface

This part of the system aims at providing a lucid user interface. Tightly bound unit of 20X4 LCD [2] [10] and 4X4 keypad [3] [10] allows the user to surf through the different menus provided by the system. The system also takes relevant inputs and outputs important messages to the user through this interface.

2.2 Sensors

There are four major types of sensors used to provide the system with inputs which are crucial for efficient and safe working of the load (motor).
1) Current Sensors,
2) Temperature Sensors,
3) Humidity Sensors,
4) Level Sensors.
These sensors have been explained in section 3.2.1

2.3 External Memory (EEPROM)

External non-volatile memory is interfaced with the controller to store the faults occurring during the working of motors. User specific data can be stored for customized application.We used Atmel's AT24CXX [4] series of EEPROM ICs for this purpose. AT24CXX series comes in the range from 1KB to 1MB storage space, each of which can be equally easily and interfaced to this system, as per the requirement.

2.4 Communication Protocol

The system can communicate with a device supporting RS232 communication protocol to transfer the data stored in memory of the system.

The definitions of '1' and '0' for RS232 communication and microcontroller are not scaled to the same voltage level. Hence, we used TTL TO RS232 level converter (MAX232) [5] as an interfacing block between the microcontroller and the PC.

2.5 Power Supply

Regulated 5V power supply through a SMPS/Linear Power Supply is needed, peak current consumption is up to 600mA.

2.6 Relay

Relay block drives the contactor which acts like as a switch for motor. Relay is driven by controller outputs.

III. WORKING

The working of the complete control system can be majorly split into two parts for explanation purpose: 1) Smart switching of pumps, 2) Monitoring of critical parameters.

3.1 Smart Switching

To ensure best possible utilization of pumps, firstly, it is necessary to decide '*how many*' pumps should be running at any given instant. If this decision dictates that only one of the two pumps is to be run, secondly, it is necessary to decide '*which one*' should be run. Both these decisions are taken by monitoring the water level via inputs to the microcontroller from float sensors. We detect three such levels using three float sensors: LOW, MID and HIGH.

3.1.1 How Many?

On detecting "HIGH" level condition, both the pumps are turned on, to draw the water out at higher rate or the reservoir might overflow. On detecting "LOW" level condition, none of the pumps will be working, as it is regarded as 'dry run' condition, hence avoiding unnecessary utilization.

On detecting "MID" level condition, only one of the twin pumps is turned on.

3.1.2 Which One?

To ensure the better utilization when only single pump is to be run, the pump that has been running for previous turn or running the most must be rested and the other one, that has been resting for the previous turn takes the charge of pumping action. To understand this better, let's imagine a situation in following order:

1) Suppose initially, the water is at LOW level and hence none of the pumps is turned-on
2) Water hits the MID level, and one of the pumps, say P1, is turned on.
3) As a result of pumping action, the water is drawn out and hence it reaches the LOW level again. Hence P1 is automatically turned off.
4) Now, suppose again water reaches MID level.
5) Now that P1 has handled the previous operation, it is the one that has been running the most, hence it is rested for the next turn and the other

pump, say P2, takes the charge of pumping action.

6) Next time water hits the MID level after falling back to LOW level or reaching the HIGH level, that is next time there is the need to turn on only one of the two pumps, it will be P1, as P2 has been active previously, hence P2 will be rested for this turn and so on.

This self-switching occurs automatically when system is put up into 'Auto Mode' by user. However, there is an option of 'Manual Mode' where user can decide which or how many pumps to be turned on manually, irrespective of the state of the level sensor inputs.

3.2 Monitoring of Critical Parameters

During both the modes, critical parameters such as current, voltage, temperature etc. are constantly monitored for protection purpose and their values are displayed on-screen. If any of these parameters exceeds the limits, which are manually set by the user, the operation of the system is suspended by turning off both the pumps and raising an alarm, hence protecting the pumps from possible damage as well as bringing the halt to the notice.The faults (over-current/over-voltage/over-heating etc.) which cause this suspension are subsequently stored into external memory and are available to be downloaded on PC via serial interface, hence easing out the debugging procedure.

3.2.1 Working of Sensors

All the sensors except for float switch (see section 3.1.1.4) have been interfaced to ADC port of the microcontroller. [1] [11]

3.2.1.1 Current Sensor (ACS712) [6]

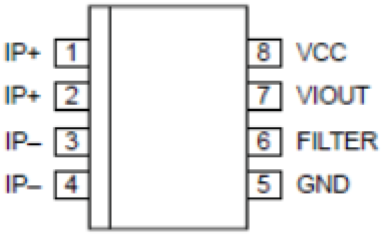


Fig 3.1 ACS712 Current Sensor Pin Diagram [6]

This sensor gives a specific voltage output corresponding to the value of current flowing through it. (100mV for 1A current for the version of sensor used in our application). It requires a supply of 5V (DC). Its Output is 2.5V when no current is flowing. Output obtained from the sensor is therefore a scaled down version of the large current flowing through main power line. It gives a sinusoidal voltage output with a mean at 2.5V, which can be given by:

$V_{OUT} = 2.5V + (0.1\, I_C)$ Volts ………. (1)

Here, I_C is the RMS AC through the line in whose series this module is connected.

This sinusoidal output is sampled at high enough rate to compute the values of the crests and the troughs with sufficient accuracy. R.M.S. value of sinusoidal voltage waveform is calculated by the formula:

$V_{RMS} = V_{P-P}/ (2\sqrt{2})$ Volts ………. (2)

This V_{RMS} is then multiplied by the scaling factor by which it was initially scaled down, which gives value of the actual current. The currents through three phases will also be sensed and will be fed to the controller unit handling any phase faults, over/under voltage or current faults.

3.2.1.2 Temperature Sensor (LM 35) [7]

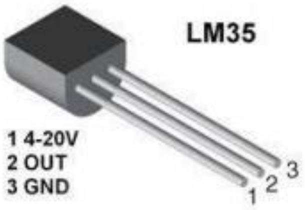


Fig. 3.2 LM35 Temperature Sensor

LM35 sensor gives voltage output 10mV per degree Celsius.

$V_{OUT}= 0\text{ mV} + 10.0\text{ mV/°C}$ Volts ………. (3) [7]

We used following configuration of LM35 in **Fig. 3.3**, which according to the datasheet [7], is implemented to measure positive temperature values only (4°C to 150°C).

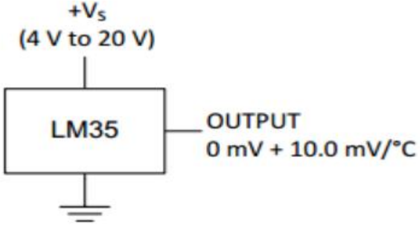


Fig 3.3 LM35 Configuration for 4°C to 150°C range [7]

This sensor is placed inside motor at location which is vulnerable to high-temperature damage.

3.2.1.3 Humidity Sensor (HRT 333) [8]

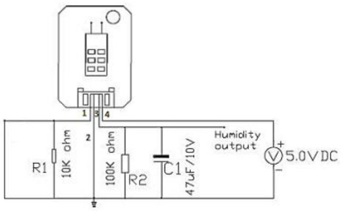
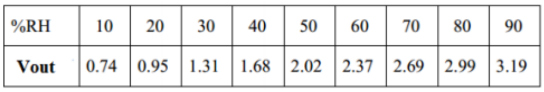


Fig 3.4 Humidity Sensor (HRT Series) Schematic [8]

Relative humidity (%RH) versus output voltage (V_{OUT}) plot is provided by the sensor manufacturer is tabulated in **Table 3.1**:

Table 3.1 %RH V/S V_{OUT} Values [8]



%RH	10	20	30	40	50	60	70	80	90
Vout	0.74	0.95	1.31	1.68	2.02	2.37	2.69	2.99	3.19

From this data in **Table 3.1**, the equation for relative humidity in terms of V_{OUT} comes out to be:

$H = (V_{OUT}-0.32)/0.034$ ………. (4)

By, inculcating the equations (1) to (4) in source code, the values of respective parameters can be found out for continuous monitoring.

Similar to the temperature sensor, this also is placed inside motor at location which is prone to humidity related faults.

3.2.1.4 Float Switch [9]

This sensor acts as a simple switch which is a single pole dual throw type switch (SPDT), having three connections viz. 'Normally Open', 'Normally Closed' and 'Common'. When water level changes the switch position, 'Normally Open' switch is closed else it is visa-a-versa.

IV. CONCLUSION

So, in our system, we targeted to reduce the cost-space footprint, enhance the programmability and upgradability of the traditional system, besides being cost effective. Also the additional features like communication protocols and intelligent operation algorithms are add-ons.

The control unit, which will be driven by a DC supply, will consist of LCD-Keypad and the controller block. So, user will have power to navigate through different options viz. Selecting Auto/ Manual mode, etc.

Hence the product will be independent and autonomous while handling the pump system. Moreover this system is adaptable to match the needs of similar loads with little or no change in the overall design.

Further up gradations like controlling the whole unit by any I.P. protocol, CAN bus/MOD bus interface can be made by using a controller supporting such protocols.

REFERENCES

Datasheets

[1]. Atmel, *ATmega32/L Datasheet-Atmel* [Online], Available: www.atmel.com/images/doc2503.pdf

[2]. Systronix (2000, July 31), *Systronix 20X4 LCD: Brief Technical Data* [Online], Available:http://www.systronix.com/access/Systronix_20x4_lcd_brief_data.pdf

[3]. Parallax, *4X4 Matrix Membrane Keypad* [Online], Available: https://www.parallax.com/sites/default/files/downloads/27899-4x4-MatrixMembrane-Keypad-v1.2.pdf

[4]. Atmel, *Two-Wire Automotive Serial EEPROM* [Online], Available: www.atmel.com/Images/doc3256.pdf

[5]. Texas Instruments, *MAX232x Dual EIA-232 Drivers/Receivers* [Online], Available: http://www.ti.com/lit/ds/symlink/max232.pdf

[6]. Allegro, *ACS712 Datasheet* [Online], Available: www.allegromicro.com/~/media/Files/**Datasheets**/**ACS712-Datasheet**.ashx

[7]. Texas Instruments, *LM35 Precision Centigrade Temperature Sensors* [Online], Available:http://www.ti.com/lit/ds/symlink/lm35.pdf

[8]. Campus Components, *HRT333 HUMIDITY SENSOR.*

[9]. Carlo Gavazzi, *Level Sensors Electromechanical Float Switches Types FS, Horizontal and Vertical Mount* [Online],Available:https://www.gavazzionline. com/pdf/FSH.pdf

Chapters in Books

[10]. Muhammad Ali Mazidi, ―LCD and Keyboard Interfacing, in *PIC Microcontroller and Embedded Systems,* 17th edition, Dorling Kindersley, India, 2008, pp. 451-477

[11]. Muhammad Ali Mazidi, ―ADC, DAC and Sensor Interfacing,‖ in *PIC Microcontroller and Embedded Systems,* 17th edition, Dorling Kindersley, India, 2008, pp. 78-480