Microcontroller-Based Single-Phase Automatic Voltage Regulator

Nang Kaythi Hlaing
Hardware Technology Department
Computer University(PangLong)
Myanmar
nangkthi.23@gmail.com

Lwin Lwin Oo
Hardware Technology Department
Computer University(Larshio)
Myanmar
lwinlwinoo.72@gmail.com

***Abstract*— This paper proposes the design and implementation of a microcontroller-based single-phase automatic voltage regulator (AVR). The basic building blocks for this design include a PIC 16f 628 microcontroller, a triac, a step-up transformer, a zero crossing circuitry and a load voltage sensing circuitry. This design is based on the principle of phase control of ac voltage using a triac. The trigger pulse for the triac is delayed by the microcontroller to provide the desired regulator terminal voltage. This voltage is always sensed and fed back to the microcontroller via a measuring unit to get a continuous control system. One of the intensions to develop this AVR is to use it in domestic heating and lighting controls. It can also be used as an adjustable voltage source by adjusting a variable resistor in the voltage sensing circuitry. It is also intended to introduce a compact AVR and to demonstrate the usefulness of the PIC microcontroller in power control field.**

Keywords-Automatic voltage regulator (AVR); PIC microcontroller; autotransformer; phase control; triac

I. Introduction

The automatic voltage regulator or AVR, as the name implies, is a device intended to regulate voltage automatically: that is to take a varying voltage level and turn it into a constant voltage level [4]. Automatic voltage regulators are widely used in electrical power field to obtain the stability and good regulation of the electric system. In typical AVRs, switching is done by electromagnetic relays, or servomotor, or electronic device, which automatically selected taps in the transformer to get the required voltage to boost (add) or buck (subtract) the input voltage. Relay tap-changers have the problems such as power lost momentarily during relay change over, unstable output and relay contact damages. Servo motor types gave the disadvantages that they have low life of the contact points of the relays [5]. Solid-state electronic device used AVRs can overcome most of the above problems as they do not use any moving part and the output voltage can vary from cycle to cycle [1][3]. For microcontrollers, they have been proven their abilities to perform well in a wide range of applications [2][6]. Nowadays, many microcontroller-based AVR can be available in markets. One of them is Sollatek AVR [7]. In this paper, it is aimed to implement PIC microcontroller-based solid sate switching automatic voltage regulator.

II. Overview Of The Proposed Design

The basic building blocks for this design include a PIC 16f 628 microcontroller, a triac, a 400V autotransformer, a zero crossing circuitry, and a load voltage sensing circuitry.

978-1-4244-5538-6/10/$26.00 ©2010 IEEE

This design is based on the principle of phase control of ac voltage using a triac, where triggering (firing) delay is determined by the PIC microcontroller. Fig. 1 shows the important components of our AVR.

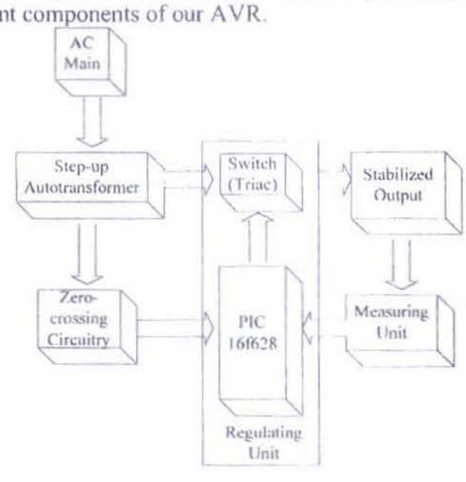

Figure 1. Block diagram of microcontroller-based automatic voltage regulator

III. Hardware Implementation And Operation Of The System Components

The general operation of the components goes like this: line voltage is stepped up to 400V using an autotransformer. Zero crossing detection circuit provides a pulse to the PIC whenever the line voltage reaches 0V. After getting the zero cross pulse, the PIC determines the delay to send a gate-trigger pulse for the triac in accordance with the output of the regulator terminal voltage. Triac, here, is used to control the phase of the line voltage. The regulator terminal voltage is always sensed and fed back to the PIC via a measuring unit.

A. Control or Regulating Unit

In control unit, PIC 16f 628 forms the heart of the unit where BTA 16(600C) triac is used as an ac switch. It is driven by PC 817 optocoupler and is triggered with a pulse sent from the PIC by taking some delay after every zero crossing of each ac main cycle. The switch is open if no trigger pulse is given to the gate. It is closed if pulses are given continuously twice every ac wave.

In our design, a 16 ampere triac is used to insure an adequate margin of safety. The triac is mounted to a heat sink which is large enough to remove the heat caused by bidirectional current flow of the triac. To suppress voltage transients a 100 Ω resistor and a 0.1μF capacitor is used as a snubber network across the triac. The presence of this network can improve the turn-on performance of the triac. In this control unit PIC 16f 628 plus PC 817 is used as a triac gate firing circuit, which is designed with care to ensure that unwanted conduction, i.e., loss of control, does not occur when triggering lasts too long. The gate-cathode resistor protects the device from false triggering due to noise.

Fig. 2 shows the interface of PIC 16f 628 with other components contained in our AVR.

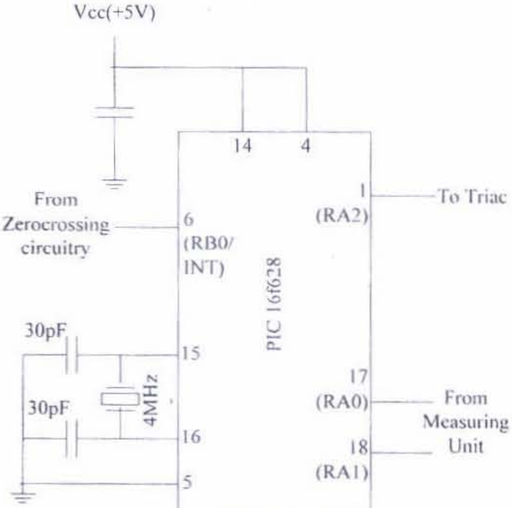

Figure 2. Microcontroller interface

B. Measuring or Feedback Unit

This measuring or feedback unit contains LM 358, the window comparator. The regulator terminal voltage is always sensed and fed back to the PIC via the circuit shown in Fig. 3.

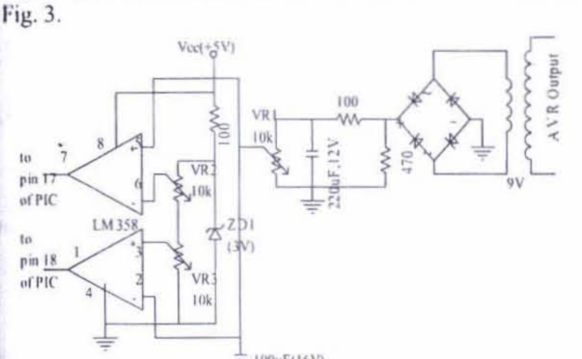

Figure 3. Schematic circuit for measuring or feedback unit

The output voltage from the AVR is stepped down to 9V and then rectified through the bridge rectifier. A 220μF, 12V capacitor and a 100 Ω resistor, which are connected in series, perform as an integrator. The values of these capacitor and resistor are chosen so that the RC value must be greater than 10ms, which is half of the period of the ac line voltage.

The feed back voltage is applied to pot VR1 and the voltage drop at its center pin is fed as a sensor voltage to pin 2 and 5 of LM 358. This sensor voltage varies proportionally in accordance with the variation of feed back voltage. The upper and lower thresholds are set by adjusting the POTs, VR2 and VR3, shown in Figure 3, which are connected in series to make ensure that the upper threshold voltage will always be higher than the lower. A 3V zener diode, connected in reverse bias with the power supply, operates as a reference for the POTs. Here, the zener voltage must be lower than the PIC power supply. But it will go into conduction when the voltage reaches or exceeds its rating. In this circuit the upper and lower thresholds are set at 1.5V and 1.38V, respectively.

As long as the sensor voltage is within these two limits, the AVR terminal voltage will be constant at 220V, and none of the two output pins of LM 358 will saturate. As a consequence, the PIC will detect no high status at its two input pins. So, it does not have to change the delay to trigger the triac which has been previously determined to provide the normal 220V. As a result the regulator retains its previous normal voltage.

C. Zerocross Detection Circuitry

Detection of zero cross point is necessary for the microcontroller to synchronize the running of its software program to the mains wave-form. This zero cross detection unit shown in Fig. 4 includes a bridge rectifier, an optocoupler (PC 817) and a transistor (C 945). This circuit accepts ac voltage signals and interfaces them to logic-level signals. Utilizing a bridge rectifier enables to sense either ac or dc voltage. In PC 817, the input side is the LED and the output side is the transistor, which are driven by different power supplies.

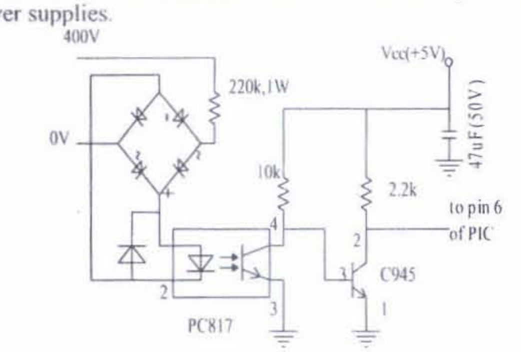

Figure 4. Schematic diagram for zerocross detection circuitry

The input side is connected to the rectified 400Vac line while the output side is connected to the PIC. A diode connected in inverse parallel to the LED on the input side of PC 817 protects the LED from damage due to over-voltage conditions.

D. Step-Up Autotransformer

A 400V step-up autotransformer having a core size of (1.5x1.4) square inches is used to handle up to 500 watt load.

For 0-400 V, 21 SWG wire is used, and for 12-0-12 V used for triac firing circuit, 23 SWG wire is used.

IV. Software Implementation for the Control Unit

The firmware program for the microcontroller is compiled with the PIC C Compiler Tool suite version 8.02 from HI-TECH Software. The source code is written in the MPLAB IDE version 6.60 from Microchip Corporation. The flow of the program, Fig. 5, goes like this: After initializing the registers, the PIC waits till the zero crossing of ac cycle reached in order to synchronize the timing of the program to the ac mains.

As soon as the PIC accepts a zero cross pulse from the zero cross detection circuit, it checks the status of the two inputs, RA0 and RA1 (pin 17 and 18). These two pins are configured as over and under voltage indicators for regulator terminal voltage. The PIC accepts the two output status of the measuring unit, which senses the regulator terminal voltage and feeds back it to the PIC.

In our AVR it is desired that the normal regulator terminal voltage to be 220V. Since, ac line voltage is typically under 220V, it is firstly stepped up to 400V to provide the desired regulator terminal voltage. So, the triac has to be triggered nearly at half of each ac main cycle to provide the desired 220V.As the TMR0 overflow interrupt is used to get the required delay for the trigger pulse, 128 is put as an initial count.

If pin 17, over-voltage indicator, is high, that is, the output voltage is greater than 220V; the PIC has to reduce that over-voltage to 220V. Thus the PIC has to take more delay to send a gate-trigger pulse for the triac than that taken to give the desired 220V. Delay is taken by decreasing the value of Trigger, which is then put into the TMR0. When the TMR0 overflows, the PIC sends the trigger pulse for the triac.

There is an under-voltage condition if pin 18, under-voltage indicator, is high; it means regulator terminal voltage is less than 220V. At this condition the PIC has to step up that voltage to normal 220V.So it has a duty to send the trigger pulse by taking less delay than that taken to get normal voltage. This step is accomplished by increasing the Trigger value.

After sending the trigger pulse for half a cycle of ac mains, the PIC again waits zero crossing for the next half cycle and follows the flow of the program as shown in Fig. 5.

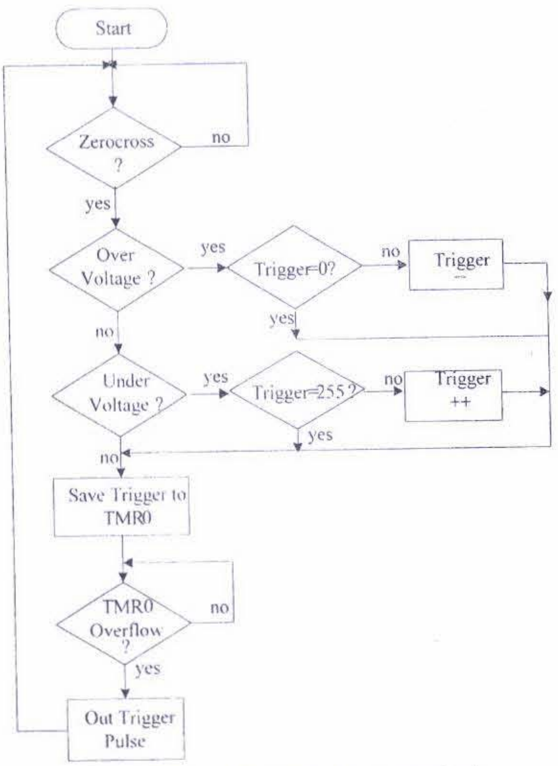

Figure 5. Software flow diagram for the control unit

V. Discussion on the Results

For our AVR, the input voltage variation or the operational range is from 170 to 240 V. This means that the triac maintaining the output voltage constant would be triggered only when the input voltage is within 170 and 240 V. In our experiments, the input voltage variation is done manually using a variable autotransformer called a variac.

Using the measurements obtained, regulation is determined by use of

$$\text{Load regulation (\%)} = \frac{V_{nl} - V_{fl}}{V_{fl}}, \quad (1)$$

where V_{nl} = voltage with no load

V_{fl} = voltage with full load.

The voltage regulation in percentage related to the input voltage variation is presented in Table 1.

TABLE I. EFFECT OF LOADS ON VOLTAGE REGULATION

AVR Input Voltage (V)	AVR Output Voltage, NoLoad (V)	Load (watt)	AVR Output Voltage, On Load (V)	Regulati-on
170	225	100	220	2.27
		200	220	2.27
		300	220	2.27
		400	215	4.65
		500	215	4.65
180	225	100	220	2.27
		200	220	2.27
		300	220	2.27
		400	215	4.65
		500	215	4.65
190	220	100	220	0
		200	220	0
		300	220	0
		400	215	2.3
		500	215	2.3
200	220	100	220	0
		200	220	0
		300	220	0
		400	220	0
		500	215	2.3
210	220	100	220	0
		200	220	0
		300	220	0
		400	220	0
		500	215	2.326
220	220	100	220	0
		200	220	0
		300	220	0
		400	220	0
		500	215	2.326
230	220	100	220	0
		200	220	0
		300	220	0
		400	215	2.326
		500	215	2.326
240	218	100	218	0
		200	218	0
		300	215	1.395
		400	210	3.809
		500	210	3.809

The input range is calculated from

$$\text{max. input range}(\%) = \frac{I/P_{max} - NM}{NM} \times 100\,, \tag{2}$$

and

$$\text{min. input range}(\%) = \frac{I/P_{min} - NM}{NM} \times 100\,. \tag{3}$$

The output accuracy is determined using

$$\text{max. output accuracy}(\%) = \frac{O/P_{max} - NM}{NM} \times 100\,, \tag{4}$$

and

$$\text{min. output accuracy}(\%) = \frac{O/P_{min} - NM}{NM} \times 100\,, \tag{5}$$

where

I/P_{max} = max. input voltage

I/P_{min} = min. input range

NM = normal voltage

O/P_{max} = max. output voltage

O/P_{min} = min. output voltage

The relationship between the input voltage variation and the AVR output on various loads can be seen in Fig. 6.

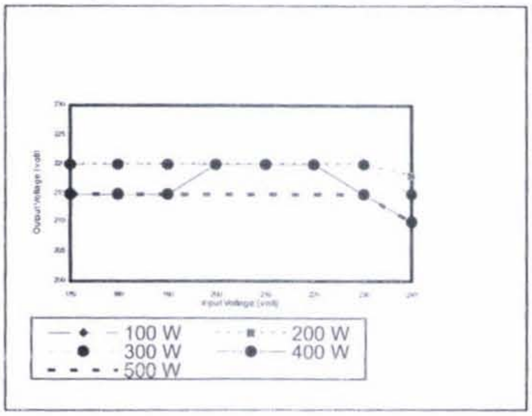

Figure 6. Input voltage vs output voltage on various load

From the results obtained, the features of our AVR can be summarized as follows: The wide input range of -22% to +9% is an ideal and essential for some places where the voltage is extremely erratic. Moreover, the output accuracy of -1% to +2.3% is sufficient for heating and lighting applications. Nevertheless, the output voltage fluctuation of our AVR is within an acceptable range.

VI. CONCLUSION AND FUTURE WORK

This paper has covered the design and implementation of microcontroller-based AVR. It is just an attempt to introduce such an AVR of having different design and operation in comparing currently available AVRs. As this design is concerned with high voltage, care has been taken in choosing suitable triac to withstand this high voltage. The uniqueness of our design is that no moving part is present and as a result, no maintenance is required. Moreover, lack of mechanical devices enables our AVR not to be encountered with disadvantages such as wear and tear of relay contact points, fatigue of the transformer taps, etc., which, but, can be found in some typical AVRs.

Our AVR, in real, can not offer a smooth regulation. This may be due to either a wide range between the thresholds of the comparator, or the core saturation of the autotransformer for the input voltage greater than 230V. Using an autotransformer with larger core size may be a remedy to solve this core saturation problem. Moreover, a better regulation will also be possible with some other peripheral modules contained in the PIC microcontroller.

Although this AVR is not a perfect one, it is hoped that this paper, at least, enables one to have better understanding on the usefulness of a PIC microcontroller in power control field.

REFERENCES

[1] A. Ahmed, Power Electronics for Technology, Prentice-Hall International, 1999.

[2] J. B. Peatmann, Design with PIC Microcontrollers, Prentice-Hall, 1998.

[3] J. M. Jacob, Power Electronics: Principles and Applications, Vikas Publishing House, 2002.

[4] M. C. Sharma, Voltage Stabilizers and Automatic Cutouts, 1987.

[5] P. C. Sen, Power Electronics, Tata McGraw-Hill Publishing Company Limited, New Delhi, 1987.

[6] R. A. Penfold, An Introduction to PIC Microcontrollers, Bernard Babani Publishing, 1997.

[7] Sollatek (UK), Automatic Voltage Regulator, www.sollatek.com