Paper No. T1-1.3, pp. 1-5

16th INTERNATIONAL SYMPOSIUM on
POWER ELECTRONICS - Ee 2011

NOVI SAD, REPUBLIC OF SERBIA, October 26th - 28th, 2011

MICROCONTROLLER BASED PHASE CONTROL CIRCUIT FOR RESONANT POWER CONVERTERS

Dejan Milcevski*, Goce Stefanov*, Ljupco Karadzinov**
*Goce Delcev University - Stip, Faculty of Electrical Engineering - Radovis, Macedonia
**Ss. Cyril and Methodius University - Skopje, Faculty of Electrical Engineering and Information Technologies - Skopje, Macedonia

Abstract: *A microcontroller based circuit is developed for driving the IGBT switches of a bridge resonant power converter used for induction heating of metals. The effect of varying the resonant load parameters due to the heating process decreases the converter output power and increases switching loses. In order to maintain constant output power, operating constantly on the resonant frequency, microcontroller program and circuit are developed. This circuit corrects the operating frequency according to the load changes by detecting output voltage to current phase difference, hence keeping the converter at resonance. The presented circuit is tested with simulations and its operation is confirmed practically with a prototype board.*

Key Words: ***Power Converter/IGBT/Phase Control/ Microcontroller***

1. INTRODUCTION

IGBT Power converter loaded with serial resonant load is very suitable for control of processes where output load changes dynamically. The induction heating process is such a process in which the load inductance changes, Fig. 1, causing the converter to operate out of resonance, hence decreasing the output power and increasing the switching losses.

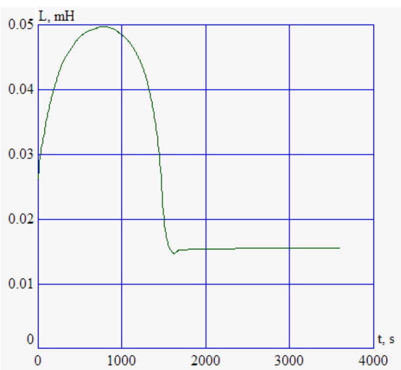


Fig. 1. *Inductance change for iron work peace for temperature change from 20°C to 900°C*

Depending on the desired mode of operation, different types of gating signals may be required for controlling the switches in the converter. IGBT switches are usually used in induction heating power converters. Since IGBTs are voltage controlled switches, the control/driver circuit should generate voltage pulses with predetermined width.

The simplest type of gating signals is when the pulse width is equal to the half of the switching period. This type of control circuit is presented in [1] and then the converter operates with constant and full power (if the load is kept at resonance). It is suitable for metal melting or thermal treatment of one or few types of metals. It is also more suitable for handlers who are more experienced with thermal treatment of metals.

In some applications the converter needs to operate with different output power in different cases. Power regulation can be achieved via regulating the output voltage or via regulating the switching frequency, [2]. The converter should operate on the resonant frequency in order to sustain zero voltage/current switching [3], and then the output power can only be regulated by varying the output voltage RMS value.

This type of power regulation has the disadvantage of generating more harmonics in the output voltage associated with more RF interferences and additional component are required for eliminating them, which also increases the converter's production costs.

In some applications of induction heating a constant temperature distribution in some areas of the work piece that is subject to induction heating is required. Such applications are induction welding of metals and thermal treatment of a particular area of a metal sample. Such applications require electronic feedback control circuit. The control circuit monitors the converter operation by measuring the output variables, voltages and currents and in case of their deviation from the defined values it acts appropriately. This type of control circuit allows the converter to operate with a constant operating point regardless on output load changes.

In this paper a design and a practical implementation of feedback control circuit with phase control is presented. The main part of the presented circuit is a microcontroller. A program code is compiled using the "C" programming language.

2. PHASE CONTROL FEEDBACK CIRCUIT

In welding application and thermal treatment of metals with induction heating, maintaining maximum output power is an imperative. In order to get the maximum power from the converter output, a control circuit must track the resonant frequency of the resonant tank. Commonly, the approach for output power regulation includes duty cycle and switching frequency regulation, [4]. The frequency control scheme is one of the most popular control schemes. However, direct frequency control results in a number of disadvantages, including high sensitivity near resonance, highly dependant from the resonant tank and requires additional control circuitry to detect operation below resonance in order to operate with zero voltage switching (ZVS) conditions. Another control method which is becoming more popular is regulation of the phase angle between the output voltage and the resonant inductor voltage or current, [3], [5]. By regulating the inductor current/voltage phase angle, the switching frequency is regulated and hence, the operating point is controlled. Phase control provides advantages of self-tuning to the tank resonant frequency (insensitive to component variations), reduced sensitivity for improved control near resonance, and inherent protection against the operation below resonance to avoid non-ZVS conditions.

2.1. Phase control algorithm for sustaining resonance at the serial resonant load in a bridge converter

The procedure for regulating the converter output power via sustaining resonance is more convenient with annulling the phase difference between the output voltage and current through the inductor (current sensing) from a practical point of view than the procedure for correcting the phase between the output voltage and inductor voltage (voltage sensing). Current sensing gives accurate results because the current is nearly sinusoidal, but requires a current transformer (CT), which is not desired for some applications. Inductor voltage sensing results in a more significant phase error and also higher harmonic components are present in the inductor voltage. In the Fig.2 the structure of the circuit for converter output power control with correcting the phase angle between the output voltage and current through the inductor is given.

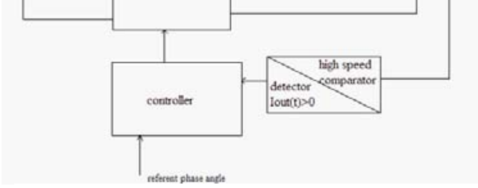


Fig. 2. *The phase controlcircuit for full bridge converter with current sensing*

An algorithm which was previously developed is used in the paper, [3]. An algorithm for sustaining constant output power of the bridge converter in mode with output loads of which inductance is changing is based on one iterative procedure for calculating the width gate pulses (switching period) for the transistors in the bridge converter.

The algorithm for output power regulation is based on:

1. Zero current detection;
2. Switching period measurement;
3. Current cycle period comparison with the period of the previous cycle;
4. Iterative procedure for determining the width of the gate pulses for IGBT transistors in the bridge.

In the Fig. 3 the flow diagram for executing the phase control algorithm that keeps the converter operation at resonance is given.

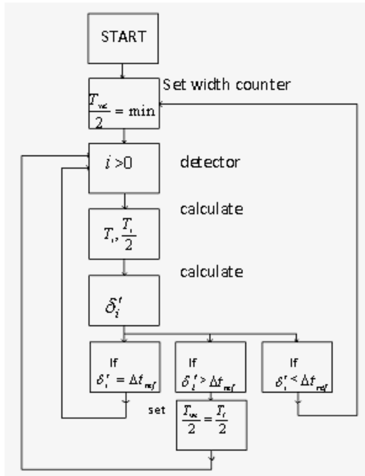


Fig. 3. *Algorithm flow diagram of the phase control circuit for the full bridge converter with serial resonant load*

When phase difference occurs according to the algorithm, with iteration in several steps the converter is brought back to resonance. The general form of control equation is, [3]:

$$Tdelay_i + \delta_i = Tdelay_{i-1} + \delta_i'. \qquad (1)$$

where $Tdelay_i$ is the period of the current cycle, $Tdelay_{i-1}$ is the period of the previous cycle, and δ_i and δ_i' are control variables.

2.2. Microcontroller based phase control circuit

This circuit is realized based on the phase control algorithm for series resonant converter described in [3]. According to the algorithm program code is developed for the microcontroller PIC18F452. The microcontroller operation with the compiled program code is simulated with the PROTEUS software on a computer. The simulation results are given in this section. Then a prototype phase control module for the resonant converter is practically constructed and its operation is experimentally verified.

The presented phase control algorithm for the resonant converter shows that by tracing the output voltage to output current phase difference, for a given resonant frequency, the converter output power can be controlled. Any deviation from the resonant frequency (caused by changing the parameters of the resonant circuit due to the dynamics of the process) is manifested by changing the phase angle between voltage and current of the converter. The algorithm proves that the iteration procedure is possible to perform adaptive tuning on the phase difference between output voltage and current of the converter, and thus indirectly to achieve a new resonant frequency which corresponds to the changes for the resonant circuit values. The algorithm also shows that it is sufficient to measure the actual cycle period and to be compared with the previous cycle period.

The control circuit utilizes a current transformer for current sensing, and then a high-speed comparator generates the square waved signal brought to the input pin of the microcontroller, Fig. 2. The algorithm is used in the practical implementation with the microcontroller PIC18F452 from the manufacturer **Microchip**, [6].

The full microcontroller based phase control circuit with the microcontroller PIC18F452 is given in Fig. 4.

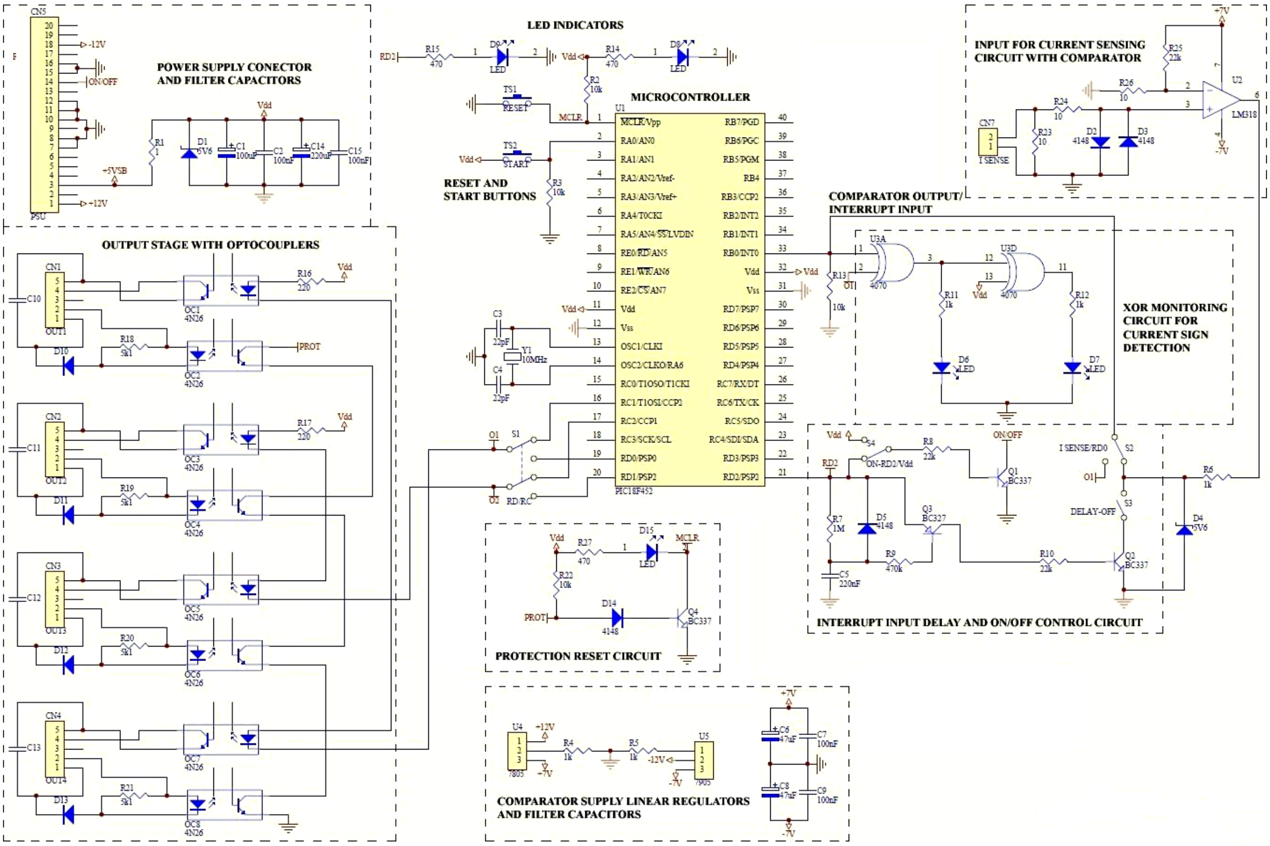


Fig. 4. *The prototype schematic diagram with PIC18F452*

The core of the phase control circuit in Fig. 4 is the microcontroller. The maximum speed of this microcontroller is 10 million instructions per second which limits its use for the phase control algorithm to approximately 40 kHz. For higher switching frequencies a faster microcontroller should be used. The current signal from the current transformer is converted into voltage and brought to the input of the circuit via connector CN7. Then it is converted into square waved signal with the comparator U2 and it is used as an input signal for the microcontroller. One output pin can be used as a START command (pin RD2 is at logical "1" while the circuit operates) and is also used for the delay function for the microcontroller input signal (if needed). There is also a monitoring sub circuit made with XOR gates. LEDs are connected to the outputs of these gates (green and red) which allow us to see if the current through the resonant load is positive and in phase with the output voltage or negative (in opposition with the voltage). The output stage consists of optocouplers driven directly from the microcontroller (points O1 and O2) in pairs. Their outputs are opposite from each other (OC1 with OC7 is the first pair and OC3 with OC5 is the second). The output connectors CN1-4 are connected to the IGBT drivers. The rest of the eight optocouplers, OC2,4,6,8, are used for feedback signals from the drivers in case a fault condition is detected from the drivers, i.e. short circuit in the bridge. The PROT point goes on logical "1" in such case and through the protection reset circuit the microcontroller stops its operation via master reset pin MCLR.

3. PROGRAM CODE SIMULATION

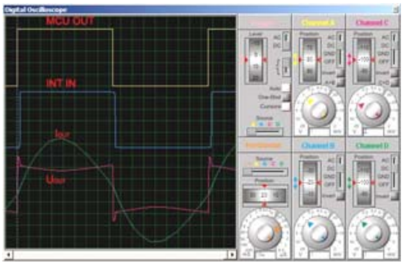
Simulations that verify the execution of the program code with the implemented phase control algorithm are done in the PROTEUS simulation program. The program code is written in the "C" programming language and then compiled into a hex file that is loaded to the microcontroller. The "C" code that was used is given in the appendix of this paper.

The simulation results are obtained for inductance changes for the values given in Table 1 (the other parameters are taken to be constant). The resonant frequency is f_o.

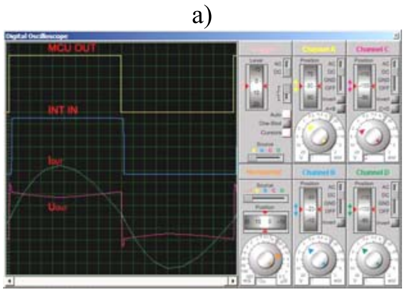
Table 1. *Values for the parameters of the simulated RLC resonant load*

R [Ω]	L [μ]	C [μF]	f_o [Hz]
0.21	26.4	26.6	6006
0.21	36.6	26.6	5101
0.21	52.8	26.6	4245

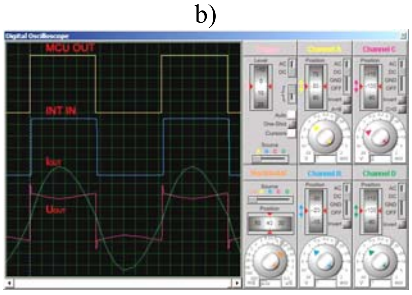
The program code can be divided into several parts. The first part is the initialization for setting the microcontroller parameters which ends with a condition command that waits the start button to be pressed (logical “1” at pin RA0). After that follows an infinite loop which generates square waved output signals with the initial switching frequency (it is the maximum frequency of the circuit and it is defined in the program code by will). These output signals are at pins RD0 and RD1 of the PIC18F452 and are used for driving the output stage optocouplers. The last part of the code is the Interrupt Service Routine (ISR) where the phase difference is detected if it occurs. The TIMER0 module is used for this action. The ISR input is the RB0/INT pin and it uses the comparator output for it. The result from the ISR is then used in the main loop for correcting the duration of the output pulses if necessary (the **while** cycle with the variable **t**).



a)



b)



c)

Fig. 5. *The results on simulation of the microcontroller based phase based circuit: a) R=0.21Ω, L=26.4uH, C=26.6uF, Time Base 10μs/div; b) R=0.21 Ω, L=36.6uH, C=26.6uF, Time Base 10μs/div; c) R=0.21 Ω, L=52.8uH, C=26.6uF, Time Base 20μs/div*

In Fig. 5 waveforms from the microcontroller circuit simulation with the given program code are shown. The waveforms are: one microcontroller output in yellow, comparator output signal (showing when the current through the resonant load is positive) used as an ISR input signal in blue, resonant load voltage in green and load current in pink. With these simulations program and hardware functionality of the circuit are verified.

The waveforms obtained with simulations show that this control circuit compensates the inductance change and keeps the resonant load in resonance. This means that the circuit operates according to the phase control algorithm when executing the given program code.

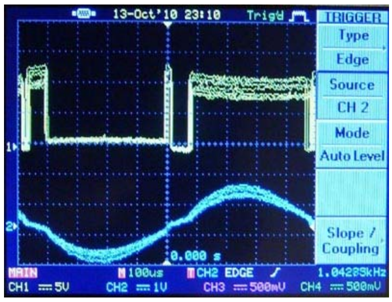
4. EXPERIMENTAL RESULTS

The prototype is tested with half bridge (it can also operate with full bridge as well) topology serial resonant converter with maximum inductance change of 28% during the experiments. Table 2 shows that the phase control circuit changes the switching frequency to adjust the resonant load circuit to the new resonant frequency.

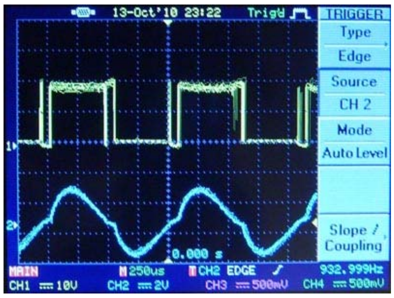
Table 2. *Values for the parameters of the RLC resonant load from the experimental results*

L [μ]	C [μF]	R [Ω]	f_o [Hz]
423	55	0.5	1043
485	55	0.5	976
526	55	0.5	932
542	55	0.5	922

The output waveforms for two different resonant frequencies are given in Fig. 6 and the prototype microcontroller board with the IGBT drivers and IGBT modules are given in Fig. 7.



a)



b)

Fig. 6. *Output voltage and current waveforms: a) R=0.5Ω, L=423uH, C=55uF; b) R=5Ω, L=526uH, C=55uF*

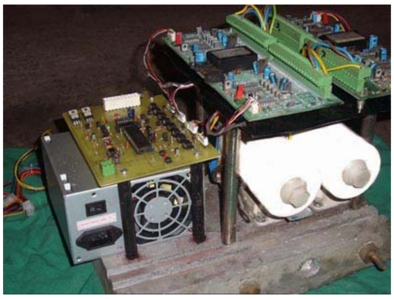


Fig. 7. *The prototype microcontroller board with the IGBT drivers and IGBT modules*

The simulations and experimental measurements show that:

- The phase control circuit simulation results are confirmed experimentally.
- With the microcontroller based phase control circuit operating with the given program code, phase differences between the output voltage and current is regulated, i.e. keeping the load at resonance.
- Converter operating with switching frequency same as the resonant frequency of the load sustains maximum converter output power.

The experimental results are for a different resonant frequency than the simulated according the technical capabilities at that time (inductor and capacitor). It is shown that the presented control circuit can operate with wide range of resonant loads with wide range of resonant frequencies.

5. CONCLUSION

This paper is a result of theoretical and experimental research of the authors for determining a solution for dynamic processes control. The presented paper solution gives the ability for adaptive control and power regulation for these processes. Induction heating of metals and heating due to dielectric losses are such processes. In these processes the dynamics is stochastic, time and space unpredictable. The device is built using a microcontroller with a quite simple code and IGBT technology. It is a simple and low cost prototype and can be used for other applications as well with simple changes in the program code.

The use of this device enables constant power converter operation, energy savings and technological processes effectuating. It can be used in metal industry, metallurgy and processing of food and wood.

6. REFERENCES

[1] G.Stefanov, L.Karadzinov, D.Milcevski: *Control Circuit for H – Bridge Power Converter with Constant Output Duty Cycle*, Annals of the „Constantin Brâncuşi" University of Târgu Jiu, Engineering Series, Issue 3/2010 Tg-Jiu, Romania, pp.189-198, 05- 11.2010.

[2] D.Milcevski, G.Stefanov, L.Karadzinov: *Design and Practical Implementation of a Variable Duty Cycle Control Circuit for Bridge Power Converter*, X Iinternational Conference ETAI 2011, Ohrid, R.Macedonia, September 2011.

[3] G.Stefanov, L.Karadzinov: *Phase Controlled Bridge Converter With Serial Resonance Load,* 14th International Power Electronics and Motion Control Conference, EPE-PEMC 2010, 978-1-4244-7855-2/10/$26.00 ©2010 IEEE, T3 81–87, Ohrid, R.Macedonia.

[4] Unver H.M., Aydemir M.T.: *Power and Frequency Control in a 60 kW Induction Steel Hesting Furnaces through PLC*, National Scientific Meetings, Ankara, Turkey, 9-12 September 2002.

[5] Yin Y., Regan Z.: *Digital Phase Control for Resonant Inverters*, IEEE Power Electronics Letters, Vol. 2, June 2004.

[6] PIC18F452 datasheet: www.microchip.com

7. APPENDIX: "C" PROGRAM CODE

```
#include <18F452.h>
#fuses H4, WDT4, NOPROTECT, NOLVP
#use delay(clock=40000000)
int t=2, k=2, d=0, s=0;
void main()   {
  setup_wdt(WDT_ON);
  setup_timer_0(RTCC_INTERNAL | RTCC_8_BIT |
RTCC_DIV_8);
  set_timer0(0);
  restart_wdt();
  output_low(PIN_D0);
  output_low(PIN_D1);
  while (!input(PIN_A0))
  restart_wdt();
  output_high(PIN_D2);
loop:
  output_low(PIN_D1);
  output_high(PIN_D0);
  set_timer0(0);
  s=1;
  delay_us(77);
  while (t>0)
  t--;
  t=k;
  restart_wdt();
  output_low(PIN_D0);
  output_high(PIN_D1);
  set_timer0(0);
  s=0;
  delay_us(80);
  while (t>0)
  t--;
  t=k;
  restart_wdt();
goto loop;
}
#INT_EXT
void presmetka()  {
  d=get_rtcc();
  if (s==1) {
    k=k+d;
    k=k-4;
  }
    else
      k=2;
  }
```