AN1114
APPLICATION NOTE

BURST MODE TRIAC CONTROL BY USING ST52x301

Authors: A. Cucuccio, M. Lo Presti

1. INTRODUCTION

The aim of this application note is to provide a complete hardware schematic and the software routines for a system using a Triac in Burst mode driven by the peripherals included in the ST52x301 DuaLogic microcontroller.

This system is suitable for thermal regulation in home appliance (i.e. heating system) or in industrial thermal controls. In particular, all the parts useful to develop the above-mentioned application are described hereafter:

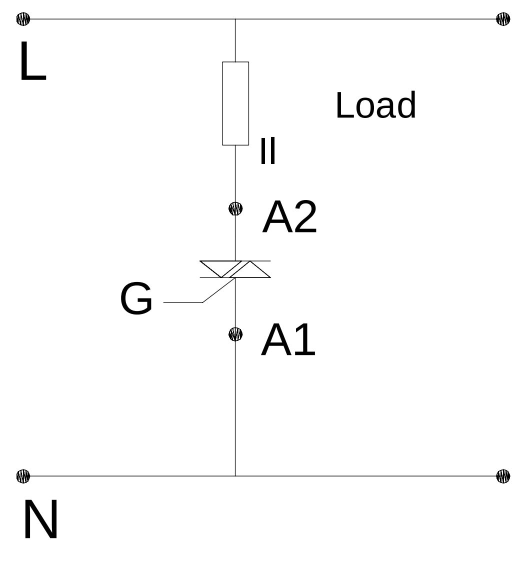
a) 1 power supply for ST52x301, from the mains voltage, in order to drive a triac in quadrant 2 and 3;

b) 1 circuit to detect the pre zero-crossing and post zero-crossing of the triac current;

c) 1 general purpose ST52x301 software to manage the application by using FUZZYSTUDIO™ 3.0 software tool.

Note: The load has been supposed as purely ohmic: in this case it is possible to directly check the main voltage to detect the zero crossing triac current.

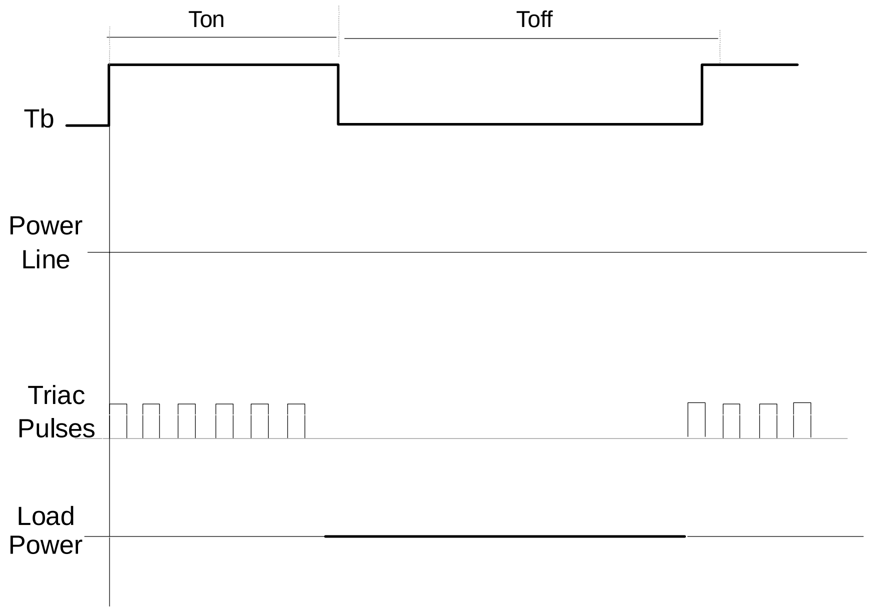
Figure 1 . Triac PinOut Notation



2. SYMMETRICAL BURST MODE CONTROL

By contrast with phase control, based upon power control on the basis of every half-wave, the periodic pulse-train control system cyclically supplies a certain number of pulses (one period=positive and negative half-wave) to the load (see following figure).

Figure 2. Periodic Pulse Train



The number of periodic pulse trains can be controlled within a given cycle time, thus determining the supplied power.

A periodic pulse train control system is used whenever the load specific time constant increases about 300 ms (heating systems with a power delay or inertia). This system is not appropriate for motor control or lighting control systems. This triac control method does not produce interferences. This means that no interference suppression measures is required.

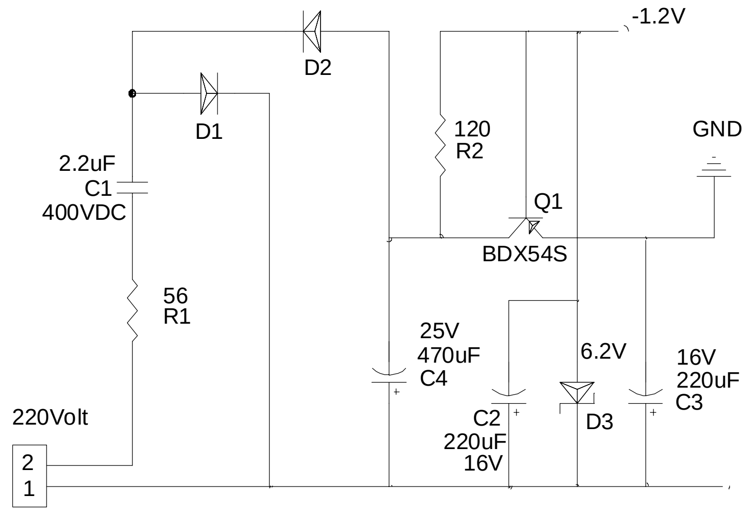
In order to implement this working mode, a flexible peripheral has been included in ST52x301 (please refer to ST52x301 Data Sheet, pages 44/51). When configured for Burst mode triac control, a square wave, Tb, (Fig. 2) is generated with a duty cycle proportional to the power the user intend to transfer on the load. A pulse is generated for each zero crossing of the main voltage included in the Ton of the fixed period.

3. POWER SUPPLY

This section, although developed as a result of a typical triac application, can be used as general purpose AC/DC Converter in any application requiring ST52x301. To fire the triac in quadrant 2 and 3, a negative polarity of G with respect to A1 is required, then a possible solution would be to keep A1 terminal to Vdd and control the gate pin with negative pulse (Vdd to Gnd transition).

The proposed system allows to obtain two outputs, -1.2V and +5V, by using above all a 6.2V Zener diode and a Darlington transistor. Using the negative output, the use of rail to rail operational amplifiers in signal conditioning circuit for the analog inputs of ST52x301 will be avoided (working range: 0V to 2.47V) thus optimizing the overall cost of the application. The described circuit is able to supply up to 50 mA with the main voltage in the range 180 - 240 Vrms. Figure 3 displays the circuit:

Figure 3. Power Supply Circuit

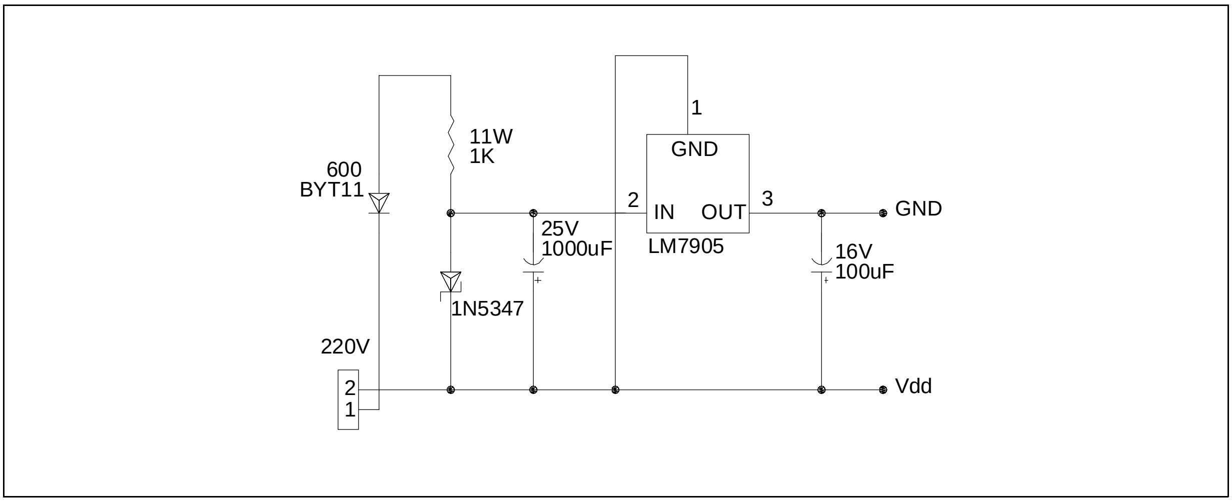


The mains voltage drops principally across C1 then we chose a capacitor with a high working voltage and a small impedence (i.e. big capacity) in order to have enough current to supply the board consumption and to hold the zener over the knee of the voltage-current characteristic. The bigger the capacitor value is, the higher the current possible to supply will be.

The zener diode provides -6.2V (as regards to the cathode of D3) to the base terminal of the darlington transistor BDX54S that is used like a voltage stabilizer. Supposing the transistor Q1 is in saturation, the emitter voltage will be 1.2V (2*Vbe=2*0.6V=1.2V) higher than the base voltage. With this configuration, it is possible to choose the emitter of Q1 for the GND, the cathode of D3 for the +5V (Vdd) voltage and the base pin of Q1 for the -1.2V voltage.

An alternative solution (but it does not provide -1.2V supply) is to use a negative voltage regulator as shown in fig. 4.

Figure 4 . Stabilized Power Supply

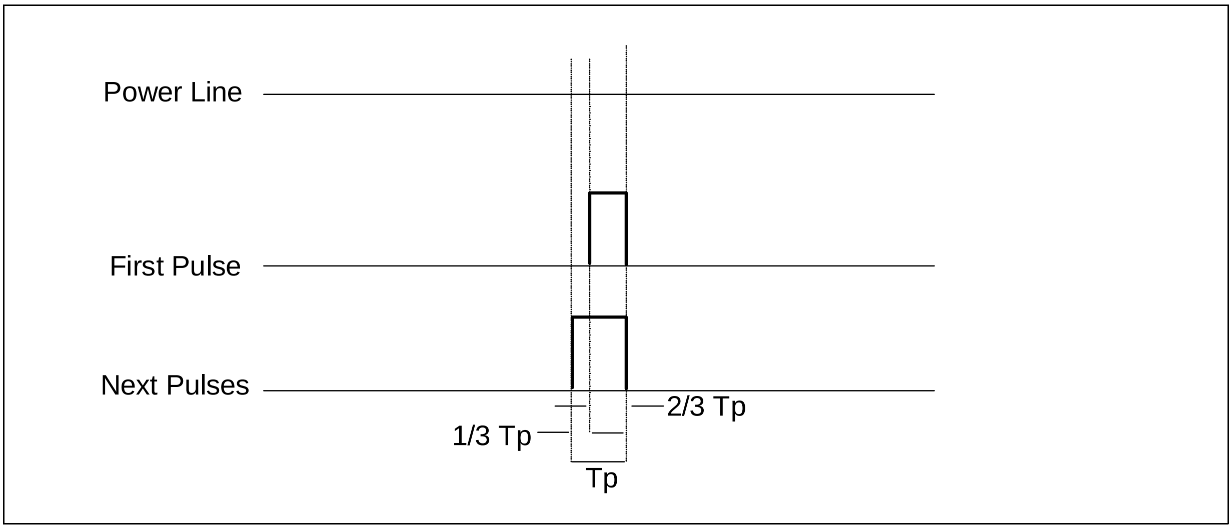


In this case it is necessary to use a power resistor rather than a big capacitor because the diode block a half wave of the mains voltage.

3.1 ZERO CROSSING

In order to work in burst mode by using the ST52x301, it is necessary to detect the pre zero crossing like shown in Fig 5.

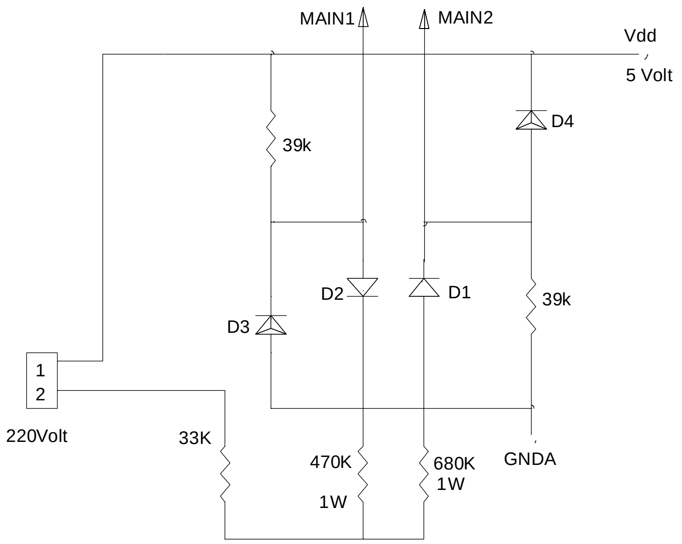
Figure 5. Burst Mode Zero Crossing



The length Tp in the ST52x301 is programmable by the user according to the characteristics of the triac. Following this approach the required triggering power can be reduced by approximately 25 %, as compared with symmetrical triggering. The first pulse is to ignite the triac whereas the following pulses are so dimensioned that the triac cannot be switched off (holding current) during the mains zero crossing. In this way, the interference signal is not generated. This allows also to avoid the problem of the flicker standard (European Standard EN60555, Part 3)

The following circuit can be used in order to obtain the previous syncronization with the main voltage then generating a sequence of pulse by the ST52x301 that will be centred on the zero crossing of the power line.

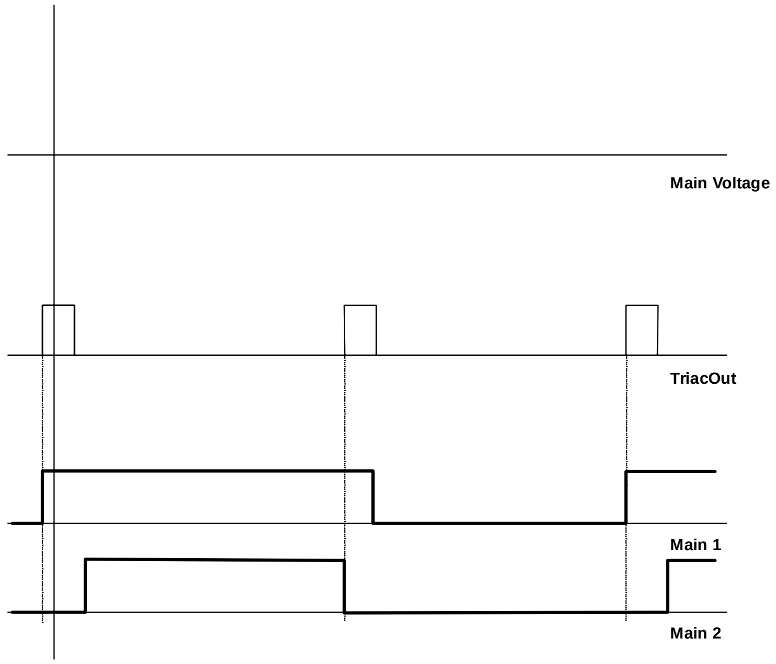
Figure 6. Burst Mode Zero Crossing Circuit



The positive and negative half-wave of the mains voltage flows through the diodes D1 and D2. When the cathode voltage of D3 (D2 anode) is lower 0.6V than the GND level, D3 clamps to 0 volt so that the low to high and the high to low transition of MAIN1 means the pre-crossing in the rising edge and the post-crossing in the falling edge respectively.

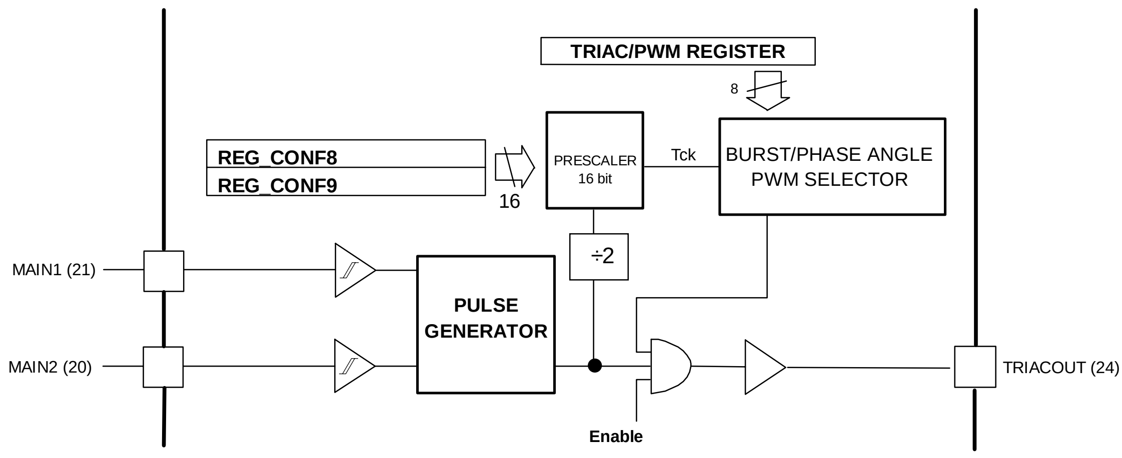
Vice versa, when the anode voltage of D4 (D1 cathode) is 0.6V higher than the VCC level, D4 clamps to 5 volt so that the low to high and the high to low transition of MAIN2 means the pre-crossing in the falling edge and the post-crossing in the rising edge respectively. The following figure summarises the previous considerations.

Figure 8 . MAIN1 and MAIN2 Wave Signals



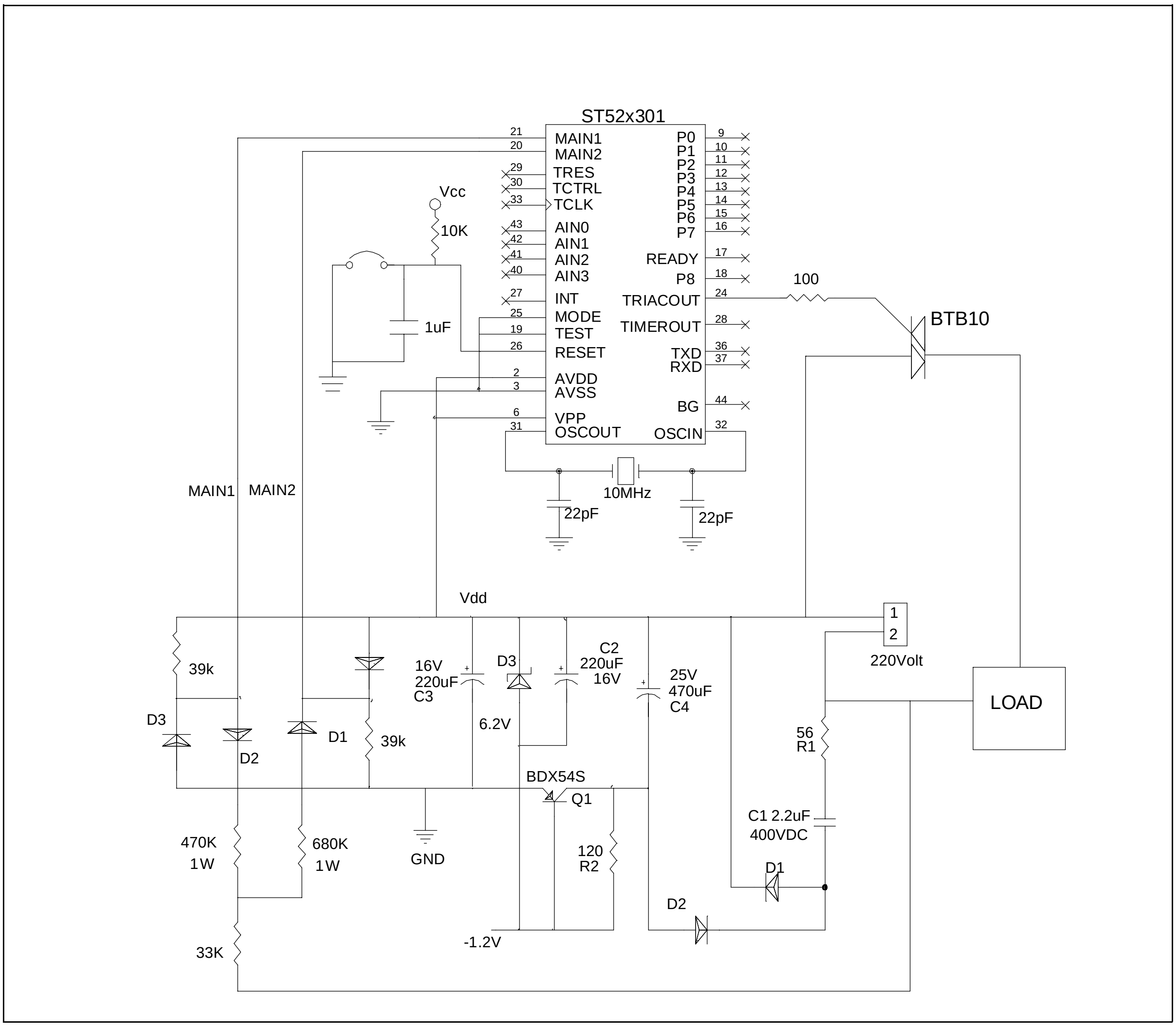
The pulses are generated by using the rise edge of the signal MAIN1 and the falling edge of the signal MAIN2. The lenght of the pulse is programmed by using ST52x301 according to the triac characteristics.

Figure 7 . Triac / PWM Driver Simplified Block Diagram



The following figure shows the complete circuit that is used in this application note for the triac burst mode control.

Figure 9. Triac Burst Mode Control Circuit



4. SOFTWARE DESCRIPTION

Once the ST52x301 peripheral for the Triac control has been programmed to drive a Triac, it is possible to set all the parameters used for the burst mode control in a flexible way.

In particular it is possible to program:

1) the period of the signal Tb: this parameter can be in the range 4.26s to 335544s according to the frequency of the main voltage (50 or 60 Hz) (see ST52x301 pag. 47 data-sheet)

2) the length of the pulse Tp: this is in the range 0.0003 ms to 19.6 ms)(see pag. 48 of ST52x301data sheet) .

3) the polarity of the pulse for the triac firing;

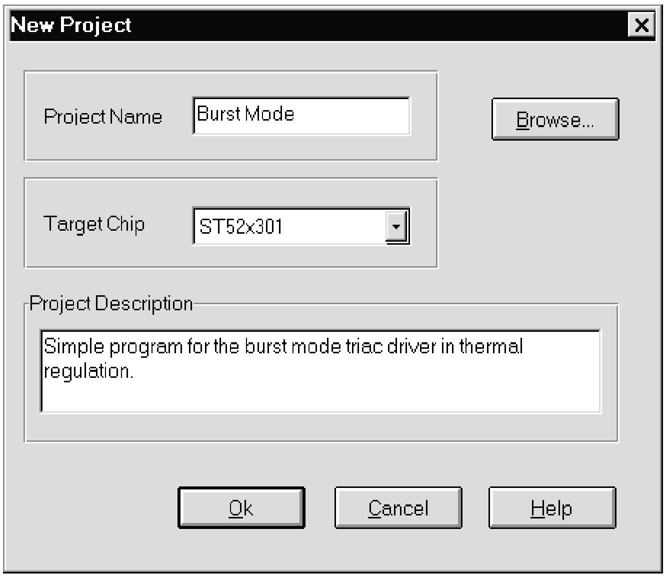
4) the Ton defining the power the user intend to transfer to the load. This parameter (8 bit) can be defined in a closed loop control for thermal regulation by using a fuzzy routine or a standard PID algorithm.

For simplicity reasons only one variable is defined in this example: the variable Ton fixing the power to transfer on the load.

Let us suppose we have a 10 MHz oscillator, then to program ST52x301:

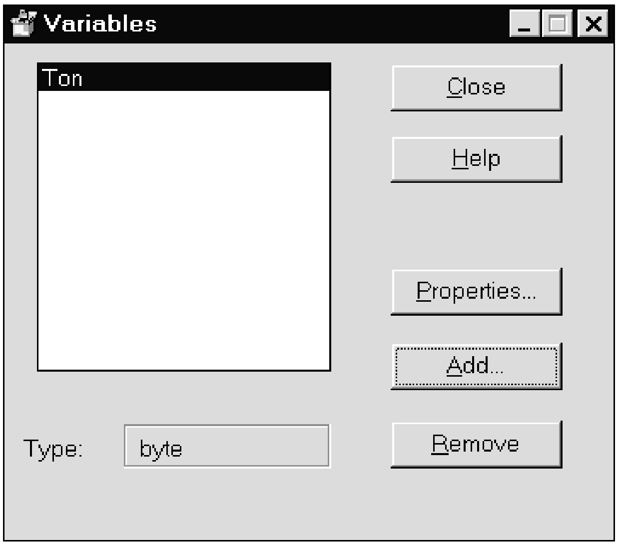
1) Open a new fs3 project (see Fig 10).

Figure 10. New Project Definition



2) Variables definition: double clicking on the variables icon, it is possible to define the global variables Ton like appear in figure 11:

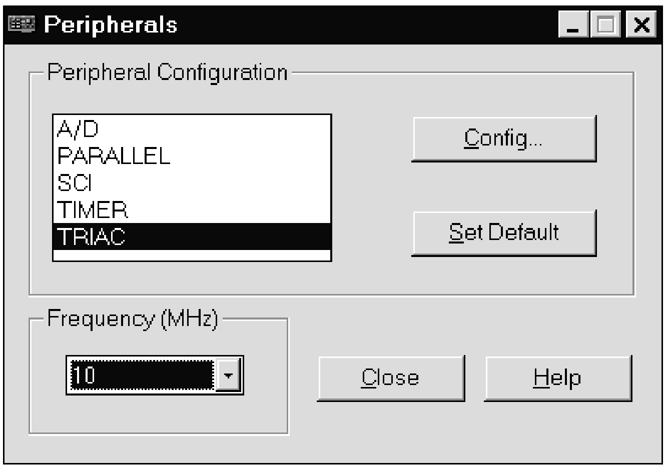
Figure 11 . Global Variable Definition



3) Configure the peripheral will be used. In this case after defining the working frequency, only the PWM/Triac driver have to be configured:

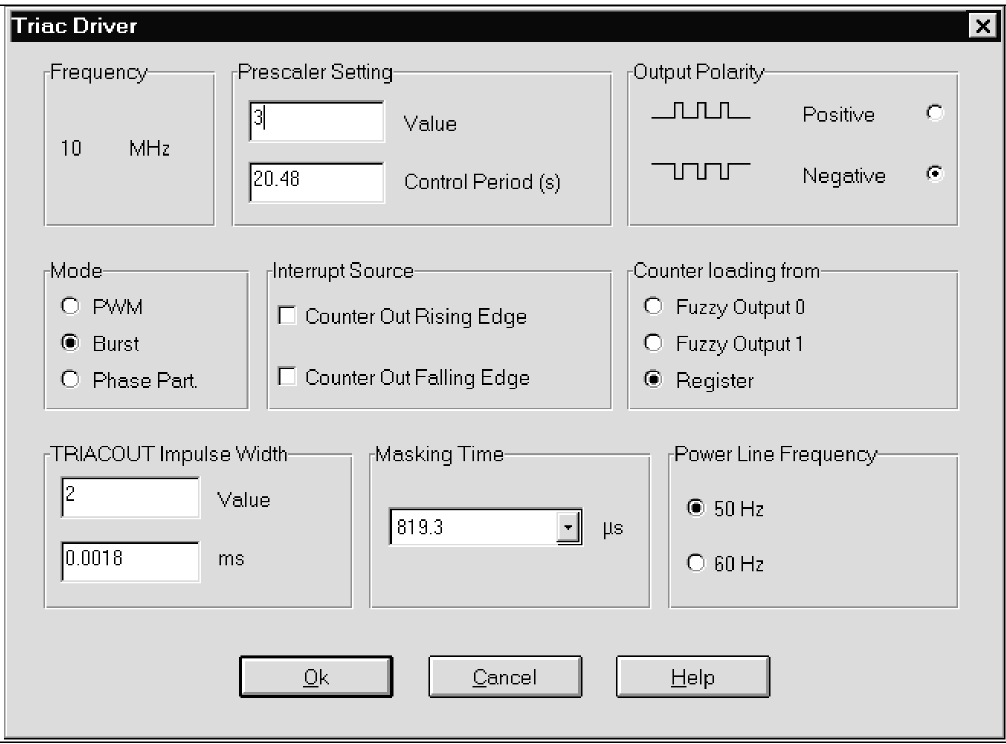
3.1) Working frequency setting: clicking on the Peripherals block it necessary to set the working frequency like shown in figure 12. (In this case a 10 MHz oscillator was selected)

Figure 12. Peripherals Configuration



3.2) Burst Mode setting: clicking on the Peripherals block and selecting TRIAC Config the following figure will appear:

Figure 13 . Triac Driver Configuration



In this window after defining that the working mode is BURST it is possible to fix all the other parameters. In particular it is possible to fix:

a) the period T: in this case it was set to 20 S;

b) the triac output polarity. In this case considering the structure of the power supply it was defined NEGATIVE to work in the 2nd or 3rd quadrant;

c) the pulse width; (0.0018 mS)

d) the power line frequency: 50 or 60 Hz;

e) the duty cycle of the burst mode can be defined by using directly a fuzzy output or by using a value stored in the Register File of the ST52x301. In this case a value coming from the Register File was selected (Ton variable)

4) Begin the project with the Main_Program block. In this window add the following blocks:

a) Arithmetic block: by using a C like istruction, set the variable Ton to 0:

Ton=0;

b) Send block (Send_Ton_to_Triac): insert the value Ton in Triac Counter to set the triac duty cycle.

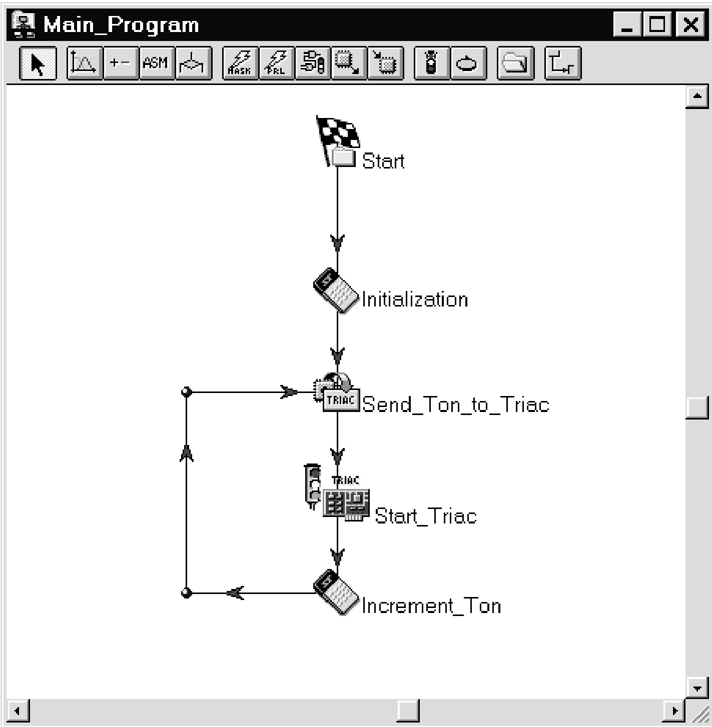
c) Peripheral block (Start_Triac): this block is necessary to initialize (set button), start (start button) the triac with the value contained in Triac Counter and enable the TROUT pin of ST52x301 (output button).

d) Arithmetic block: (Increment Ton) this block was included just to show a simple operation to perform the Ton values. In this case, a simple increment of 1 unit is considered in each cycle.

Ton+=1;

Normally this value is defined by a fuzzy routine or by a standard algorithm for closed loop control.

Figure 14. Main Program Description



APPENDIX

In appendix is shown the assembler code to implement the previous application.

```
;   Source file:                        C:\ANBM\BURST.wcl
;   Compile time:       Wed Sep 30 14:27:01 1998
;   Device type:        ST52x301
;   Compiler version:   01.00 (02.06.98)

@@WCLStart@@:
        ldcf        0       0
        ldcf        1       2
        ldcf        2       6
        ldcf        3       0
        ldcf        4       0
        ldcf        5       0
        ldcf        6       40
        ldcf        7       0
        ldcf        8       3
        ldcf        9       0
        ldcf        10      104
        ldcf        11      1
        ldcf        12      64
        ldcf        13      2
        ldcf        14      0
        ldcf        15      228
Start:
Initialization:
        ldrc        15      0
Send_Ton_to_Triac:
        ldpr        1       15
Start_Triac:
        ldcf        11      3
        ldcf        10      107
Increment_Ton:
        mdgi
        ldrc        0       1
        add         15      0
        megi
        mdgi
        ldrc        0       255
        sub         0       15
        megi
        jpns        @@00000
```

```
@@00001:
        ldrc        15      255
@@00000:
@@00002:
        jp          Send_Ton_to_Triac
        stop
```