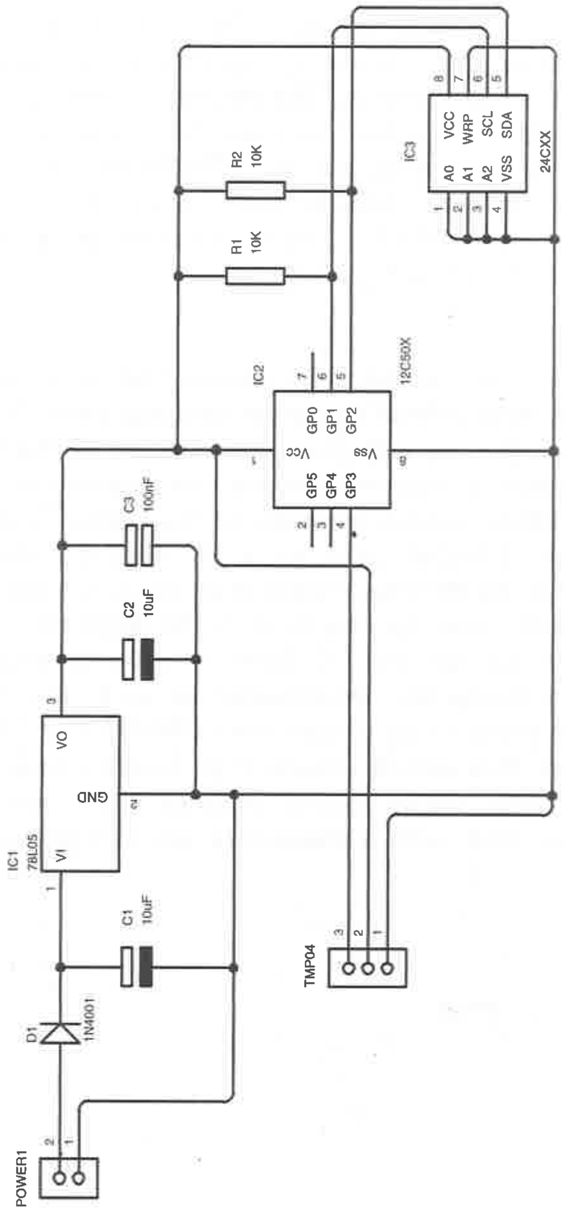

FIGURE 6

Mains Powered Projects

DON'T DO IT! There is the disclaimer. Playing with mains is for those that do not like this life. As a general rule, if it can not run from a 9V battery then give the project to your junior engineer. If you are that junior engineer then read on.

Many of the code modules have ended up driving a relay or transistor as the power switch. We also have the option of switching mains voltages at many amps with a triac. A triac is a switching element that will conduct current in both directions and will continue conducting as long as there is a minimum current flowing through it. This means that once switched on, it will not switch off until no current is flowing through it. This makes it a great switch for AC circuits as it will switch off as the current changes from one phase to the other. To drive this device we only need to pulse the Gate pin for a few microseconds until the current through it is sufficient to keep it open. However, there are a few practical considerations to keep it mind when designing such a system. Triacs come in a range of voltage and current ratings, but for most applications a TO220 packaged devise will give you a 600V switch capable of switching currents up to 8 Amps with only a modest heatsink. A Typical part number would be BTA08-600. The important specification for the triac is the gate current required to switch the device on. This is always given in milliamps and specified in 4 quadrants. Consider the circuit diagram in figure 7. It shows a capacitor/resistor dropper to generate a 5V supply and shows also how we would drive a triac. Note that the capacitor MUST be a X2 400V rated cap and that the resistor will have to have a power rating of a couple of watts. Lets consider two of the pins, the gate and A1. We can drive the gate with a positive or negative signal with respect to A1 to switch the triac on and pin A1 can be positive or negative with respect to A2. So we have 4 different conditions for switching on the device. When the current is measured we find the following as being typical :-

Device BTA06-600D

A1	A2	G	switch on current
+	-	+	10mA
-	+	+	5mA
+	-	-	5mA
-	+	-	5mA

From the above we can see that our power requirements will be minimised if we drive the gate low with respect to A1. This is why we have the +5V rail on the PIC connected to the Live side of the mains. **SO BE CAREFUL!**

The power supply for this arrangement is derived directly from the mains. There is no isolation so DO NOT plug your emulator in without isolating it from the mains.

That's all the DO NOTs' out the way so lets consider what we can do with such a circuit.

Lets start by looking at a simple switching circuit based on the diagram in figure 7.

We can monitor the mains cycle on pin GP4. Do not use GP3 as this is the one pin that does not have blocking diodes to clamp the input voltage. We could end up with more than 13V on this pin and end up reprogramming the part. When the live line is negative with respect to neutral we will have a low on GP4 and when it is positive we will have a high. So we can determine the zero crossing point of the mains with some accuracy. This is the point (or shortly afterwards) that we want to switch our load. This will minimise RF interference caused by the switching as only very small currents are being switched. It will also maximise the life of the device that is being switched for the same reason. This would be the route to everlasting light bulbs. To wait for the zero crossing point we could call a subroutine that looks for a transition on the mains signal before returning.


```
WAIT_FOR_MAINS_ZERO
      BTFSC   MAINS_IN
      GOTO    MAINS_HIGH

MAINS_LOW
;we wait in this loop for mains to go high
      BTFSS   MAINS_IN
      GOTO    MAINS_LOW
      RETLW   0FFH

MAINS_HIGH
;we wait in this loop for mains to go low
      BTFSC   MAINS_IN
      GOTO    MAINS_LOW
      RETLW   0FFH
```

When we returned from the above subroutine we would then want to switch on the triac. We need only use the BCF instruction to do this. The downside of such a simple scheme would be the unnecessary current drain through the gate pin. So we pulse the gate until we are sure that the gate is open and will not close until the next zero crossing.

```
PULSE_TRIAC
      MOVLW 005H
MOVWF COUNT   ;number of firing pulses
PT1
      BCF     TRIAC
      NOP
      BSF     TRIAC
      CALL    V_SHORT_DELAY
      DECFSZ  COUNT
      GOTO    PT1
      RETLW   0FF
```

The routine V_SHORT_DELAY will determine at what point on the mains cycle we stop firing the triac. This will be influenced by the holding current of the triac, or that current which MUST be flowing through the triac for it to stay open. This is typically of the order of a few tens of milliamps but check the data sheet of the part being used.

cycle takes.

that when we start switching at points other than the z ing point we have to include a large choke in line with the t to cut down on RF noise.

```
;let us set TMR0 to count inst.                e prescalar set for /64
      MOVLW  B'xx000101'   ;p
      OPTION

      CLRF      TMR0
      CALL      WAIT_FOR_MAINS_ZERO
      MOVFW  TMR0
;when we get here W now contains a number that represents
;one half cycle of the mains
      MOVWF  MAINS_TIME

;just to ensure that we do not miss the zero point
      DECF      MAINS_TIME

SOFT_START
      DECF      MAINS_TIME
```

Mains Powered Projects
THE GREATEST LITTLE PIC BOOK

```
;lets stay at each level for 4 half cycles
      MOVLW  04H
      MOVWF  COUNT1

SS1
      CALL      WAIT_FOR_MAINS_ZERO
;lets start the timing here
      CLRF      TMR0
SS2
      MOVFW  MAINS_TIME
      XORWF   TMR0,W
      BTFSS     Z            ;skip if we have a match
      GOTO      SS2          ;loop back if no match
;if we get here then fire the triac
;the PULSE_TRIAC routine must be finished
;before we cross the next zero point
      CALL      PULSE_TRIAC
      DECFSZ   COUNT1
      GOTO      SS1
;when we get here we have held the power level for 4 half cycles
      DECFSZ   MAINS_TIME
      GOTO      SOFT_START
;when we get here we are at full power
SS3
      CALL      WAIT_FOR_MAINS_ZERO
      CALL      PULSE_TRIAC
      GOTO      SS3

      END
```

Please note that the above routines have not been tested on a live rig but only on a 12V system.

An alternative to switching during each cycle is to 'cycle steal'. This

a minimum.

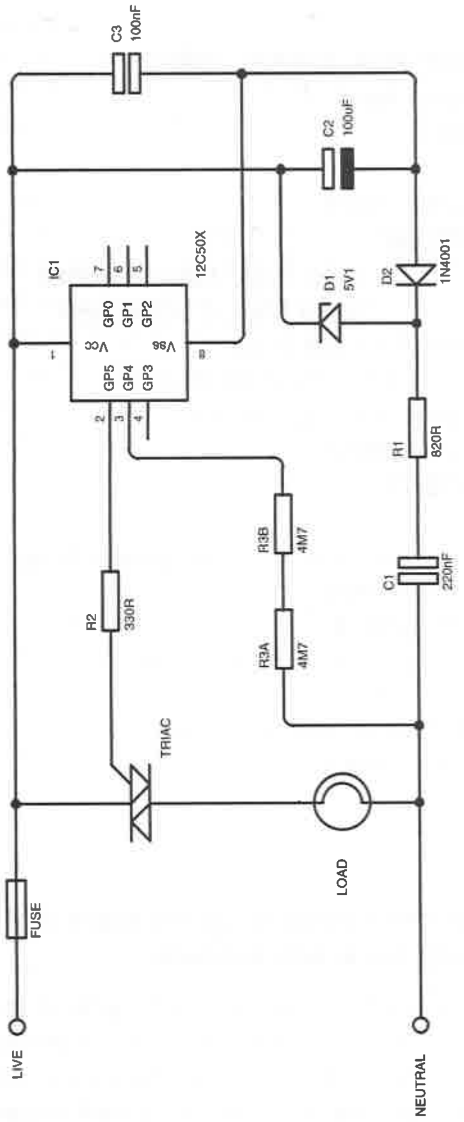

FIGURE 7

On the subject of AC power control, consider your old valve amplifier (or your new Quartz Halogen desk lamp). The valves all have a heater voltage of 6.3V AC. The failure mode for most valves is the inrush current at switch on blowing the heating element. If we give it a soft start we can look forward to valves that last forever (almost). Now there are at least two ways of achieving this control. The first is to use a circuit similar to that above. The down side of such a circuit is that we will loose approximately 2V across the triac. This is not a problem on a 240V system but a major loss on a 6.3V system. The alternative is to reference the PIC 0V line to the most negative part of the AC signal and use either a standard NPN transistor or a power MOSFET. When we are at maximum power we switch on the relay (just for the HIFI purists) and put the controller to sleep to minimise any clock noise. The power is controlled using Pulse Width Modulation (PWM) to switch the transistor on and off. The more ON time the power to the heaters. Lets look at some code segments for this job :-

```
;make sure that the WDOG timer is off
COLDSTART
;wait here for a few secs and then setup I/O
      CALL      LONG_DELAY

      MOVLW  B'0010x0xx'    ;GP2 = RELAY
                            ;GP4 = MAINS IN
                            ;GP5=TRANSISTOR DRIVE
      TRIS      GPIO
;lets set TMR0 to count inst. cycles with the prescalar set for /64
      MOVLW  B'xx000101'    ;prescalar of 64
      OPTION

;lets set up our counters
      MOVLW  01H
      MOVWF  COUNT1

;ensure we start at the next zero crossing
      CALL      WAIT_FOR_MAINS_ZERO

LOOP0
```

```
      MOVLW  018H
      MOVWF  COUNT3

;on loop
LOOP1
      MOVFW  COUNT1
      MOVWF  COUNT2
LOOP2
      BSF    TRANSISTOR
      DECFSZ COUNT2
      GOTO   LOOP2

;off loop
      COMF   COUNT1,W
      MOVWF  COUNT2
LOOP3
      BCF    TRANSISTOR
      DECFSZ COUNT2
      GOTO   LOOP3

      DECFSZ COUNT3 ;just to lengthen ramp

;now lets change the on/off ratio
      INCFSZ COUNT1
      GOTO   LOOP0
;if we get here then we are on 100% so
;switch in relay and go to sleep
;this will minimise noise and power losses
      BSF    RELAY
      SLEEP
;we will never wake from sleep and only restart
;when power is removed at switch off then switched on again

      END
```


Further Reading

PIC12C5xx Data Sheet DS40139A	Microchip
Non-Volatile Memory Products Data Book	Microchip
PIC16/17 Microcontrollers Data Book	Microchip
The Embedded Control Handbook	Microchip
The ECHB supplemental 1	Microchip
A Beginners Guide to the Microchip PIC	Bluebird Technical Press
PIC COOK BOOK 1	Bluebird Technical Press
PIC COOK BOOK 2	Bluebird Technical Press
A Practical Introduction To Electronics	Bluebird Electronics