



MProTimer

2010

Eric Rodda

Kestar Enterprises
www.kestar.com.au

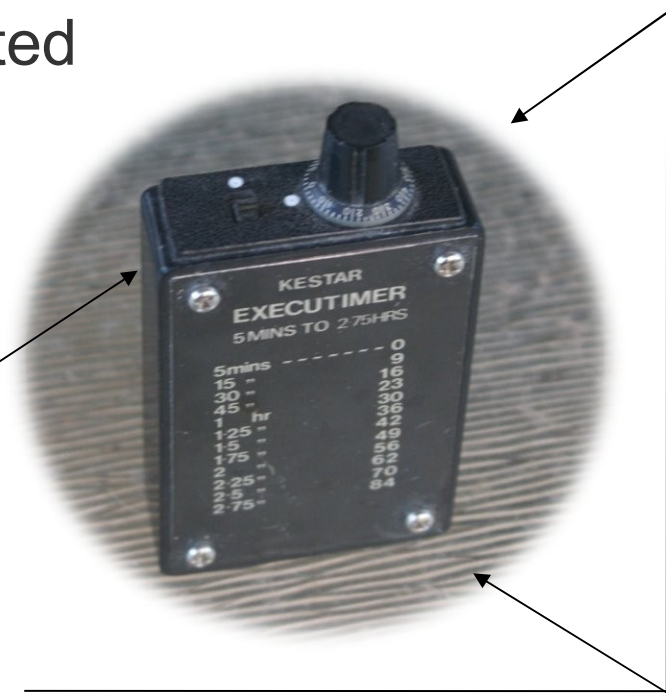
Requirements of the MProTimer

- The device had to be small. Pocket size!
- It had to be easy to use.
- It would be nice if it was easy to produce.
- Functionality is important
 - will it do everything required ?



History

- The predecessor of the MProTimer was the **EXECUTIMER**
 - This was small
 - It was not easy to make.
 - It did nearly everything I wanted
 - ...but it had problems.



- It had an ON/OFF switch
 - you had to remember to turn it off.
- It used a chart to obtain a number, then the timing knob was set to this number. This knob could be knocked.

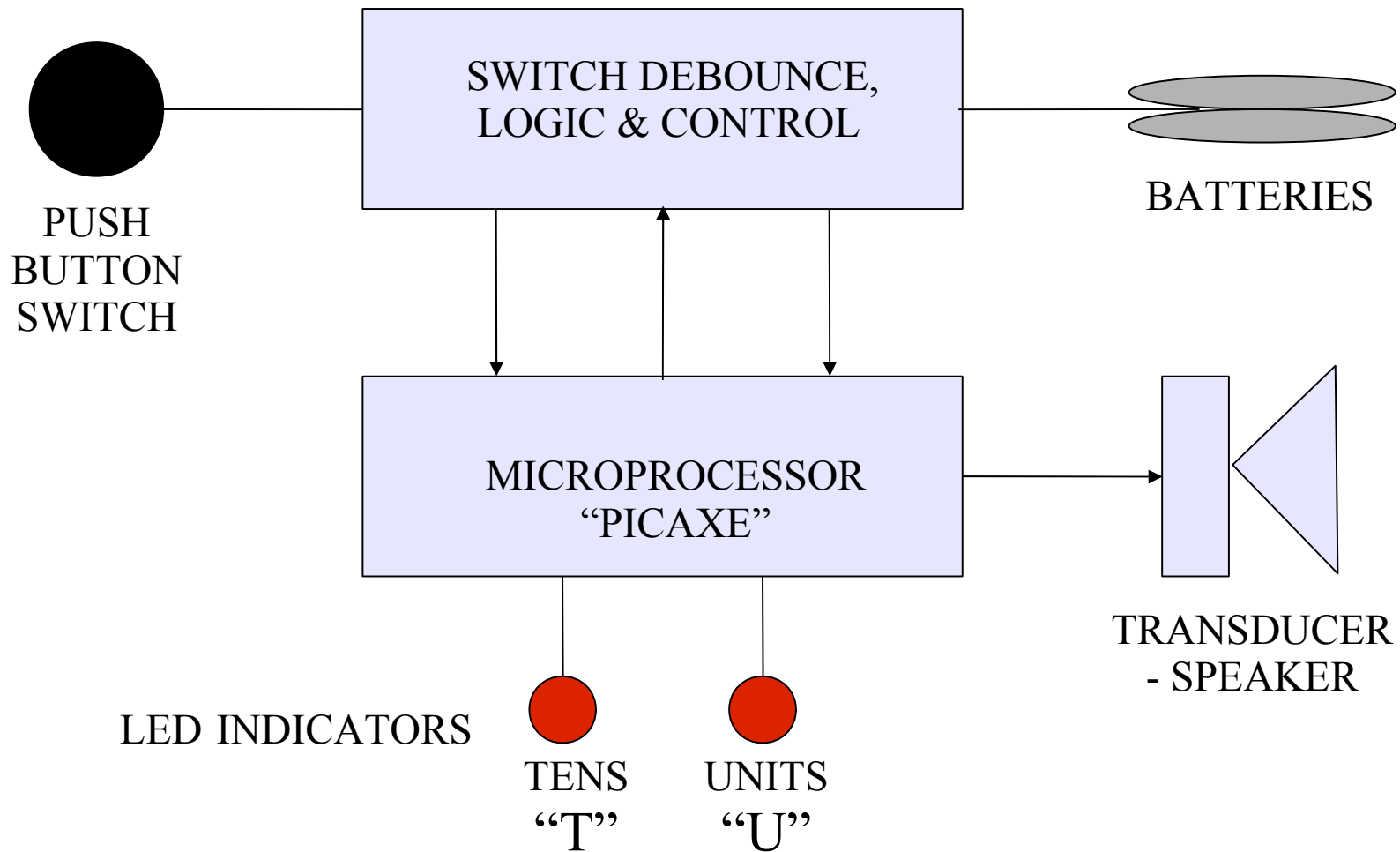
Enter the MProTimer ...

Features

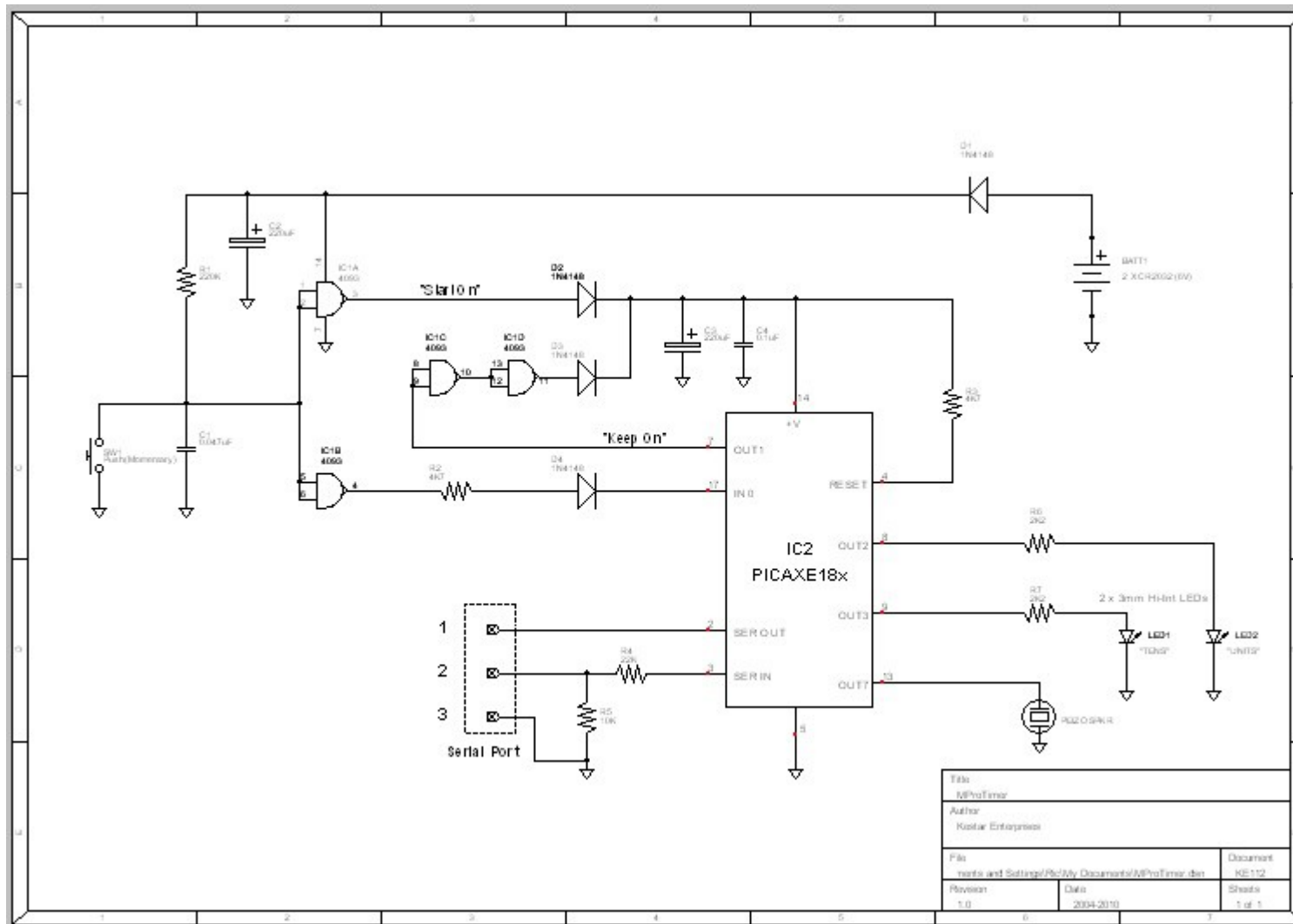
- Small ? - Yes
- No ON/OFF switch (as such) to forget... It turns itself off.
- No knob to knock !
- Microprocessor controlled, which allows for program changes and multiple functions with one button.
- Relatively easy to produce.



What makes up the MProTimer ...



What makes up the MProTimer ...



...

What makes up the MProTimer ...

- Plus 130 lines of computer code !

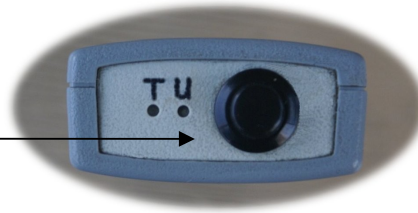
```
19 main:
20 if SW = 1 then solidpush
21     goto main
22 solidpush:                                'ensures switch is actually pushed, not a glitch
23     pause 60                               'wait a bit
24     if SW = 1 then letsbegin                'check switch again
25     goto main
26 letsbegin:
27     sound SOUNDER, (200,1)
28     high POWER                               ' Keep on signal
29     let b1 = 0
30     let b2 = 0
31     let b6 = 0
32     wait 1
33     'Capture tens of minutes
34     high LED1
35     count 0,5000,b1                          ' count for 5 seconds
36     low LED1
37     if b1 <= 9 then min10ok
38     let b1 = 9 'make it 90 mins max
39 min10ok:
40     'Capture units of minutes
41     high LED2
42     count 0,5000,b2                          ' count for 5 seconds
43     low LED2
44     if b2 <= 9 then minsok
45     let b2 = 9 'make it 9 mins max
46 minsok:
47     ' check if no time setting selected and exit if so.
48     if b1 = 0 and b2 = 0 then cancelit
49     gosub display
50     let b1 = 0
51     let b2 = 0
```

...section of code displayed

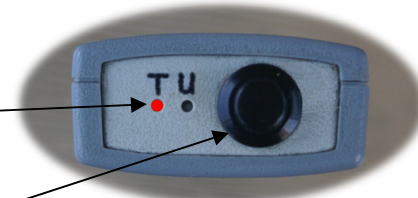
How does the MProTimer work ...

Timing example: 23 minutes.

1. Press the button to turn it on

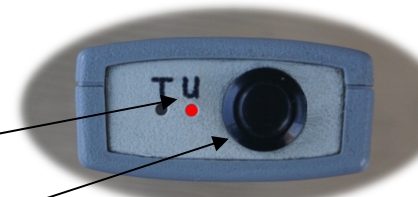


2. The Tens of minutes LED lights



3. Press the button: one press for each ten minutes. Eg. 2 presses

4. Wait for the Units LED to light



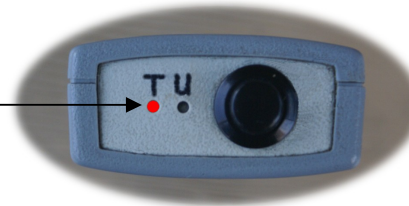
5. Press the button: one press for each minute. Eg. 3 presses

The unit is now programmed for 23 minutes and timing has begun.

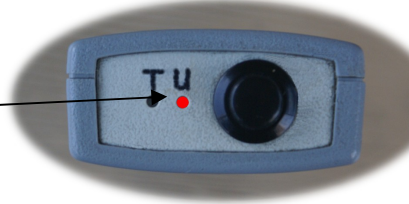
How does the MProTimer work ...

Timing example: 23 minutes.

6. The Tens LED will now flash slowly.



7. In the last five minutes the Units LED will flash slowly.



8. During the last minute, the Tens and Units LEDs flash alternatively.

9. During the last 10 seconds, the unit makes a clicking sound, with both LEDs flashing warning that the time is nearly up.

Click, click,
click...

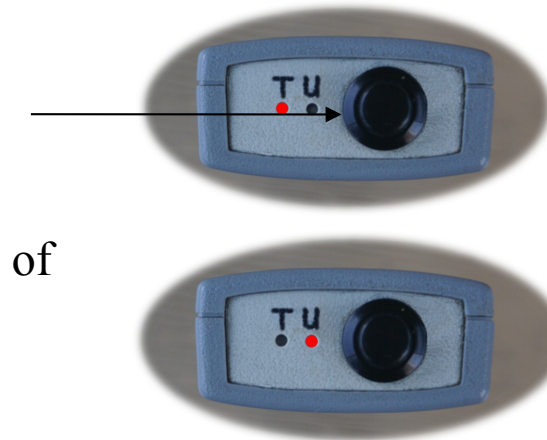
9. If the button is not pressed the unit will then make the alarm sound.

Beep, beep, beep...

How does the MProTimer work ...

Timing example: 23 minutes.

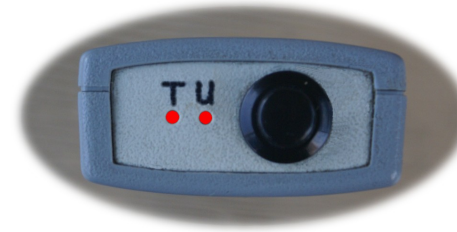
10. If the button is held in during the timing period the unit will cease timing and display the remaining time on the LEDs, flashing the Tens LED with the number of tens minutes and then the Units LED shows the remaining minutes.



How does the MProTimer work ...

The unit now has another timing feature...

After turning the unit on, if the button is not pressed while the Tens or Units LEDs light, the unit will light both LEDs.



If the button is then pressed while both LEDs are lit, the unit goes into count-up timer mode so that an event etc. can be timed.

At the end of the event, the button is pressed and the LEDs show the elapsed time. The number of flashes from the Tens LED represent the tens of minutes and then the Units LED the number of minutes.

Eg. Five flashes from “T” equals 50 minutes and six flashes from “U” equals 6 minutes. Total 56 minutes.

QUESTIONS ??

**Original description and circuit published in Silicon
Chip magazine
May 2006 Page 36...**

**The circuit diagram, program and
document are available to be freely
downloaded
from
www.kestar.com.au**

Thank you

**Eric Rodda
Kestar Enterprises
www.kestar.com.au**