



**Ron Russa** is this month's winner of the Peak Atlas ECH Meter!

### 6-station sprinkler controller

This design makes use of a cheap quartz clock mechanism along with a PICAXE-08 micro to accurately time a 6-station sprinkler system.

Power for the entire setup is sourced from a 12V DC plugpack. A 7805 regulator (REG1) provides a well-regulated +5V for IC1 & IC2. Two resistors divide the +5V rail down to about +2.7V to power the clock board. If desired, a 3-cell alkaline battery pack and series diode (D2) can be included to provide backup power during short-term outages.

The quartz clock board acts as a timebase for the sprinkler system, eliminating the inaccuracies that arise when using the micro's on-board resonator and software for timing tasks. Accuracy of a typical clock is around 2ppm, which means only about one minute error per year!

The existing coil in the clock mechanism is first removed and discarded-

ALTERNATIVE METHOD OF DRIVING SOLENOID VALVES VIA RELAYS

ed. The two pads are then wired to Schottky diodes D1 & D2. These diodes "OR" the pulses from the clock circuit, resulting in a 30ms pulse to pin 4 of the PICAXE microcontroller every second. A LED on pin 3 of the micro flashes in unison with the tick of the clock to indicate that the program is running.

On the output side, each solenoid is switched by a MOSFET, which is in turn controlled by one output of a 4028 1-of-10 decoder (IC2). The decoder enables all six stations to be switched using just three port pins of the PICAXE.

Let's now look at the PICAXE program, as shown at right. Because of code space restrictions, the six stations are programmed within a 2-hour time zone, operating three times a week. The use of 2-hour blocks for timekeeping cuts down on the number of variables and therefore the required code. This allows more sprinklers to operate on the same variable than would otherwise be possible and allows a maximum watering time of 20 minutes per station. The program uses 126 bytes out of the 128 available.

System activation time is determined by the initial value of variable b3, which counts two-hour time periods. If b3 = 0 (the default), then the first sprinkler will start at power-up. On the other hand, if you wanted to power up the system on, say, Sunday at 12.00 noon, but would like to start sprinkling on Tuesday at 8.00pm, then b3 should initially be set to the value 28 ((24hrs + 24hrs + 8hrs)/2).

If you only needed to water twice a week, you would delete the last "if b3 =" instruction and adjust the second b3 value, as so on. It's all pretty straightforward and is easily modified to suit your requirements. The programmed sequence will repeat indefinitely until power is removed or a new program is loaded.

As it is very easy to download a new program to the PICAXE, changing sprinkling times as needed for summer and winter is a snap. This eliminates the need for menu-driven push-buttons and therefore complicated hardware. And best of all, it's much cheaper than commercial units, which typically cost over \$120!

**Ron Russo,  
Kirwan, Qld.**

#### 'Six Station Sprinkler Controller - PICAXE-08

\* NOTE: Values shown for b3 (minutes variable) are for maximum allowable time of 20 minutes each but they can be any length as long as the total time for the six does not exceed 119 minutes. If less stations are needed, then change the program or just put the same last time into redundant stations.

Symbol Work\_Led = 4  
dirs = %00010111

'pin for indicator LED  
'set for 4 outputs

Main:  
if pin3=1 then Sec\_Pulse  
low Work\_Led  
goto Main

'is Pin3 is high goto Sec\_Pulse  
'if not then turn LED off

Sec\_Pulse:  
high Work\_Led  
pause 50  
let b1=b1+1  
if b1=60 then Min\_Up

'a pulse so turn LED on  
'wait 50 ms (30 ms clock pulse ended)  
'increment seconds  
'if 60 seconds are up goto Min\_Up

Chk\_Day:  
if b3=0 then Station\_1  
if b3=24 then Station\_1  
if b3=48 then Station\_1  
goto Main

'if time first day goto Station\_1  
'if time second day goto Station\_1  
'if time last day goto Station\_1

Min\_Up:  
let b1=0  
let b2=b2+1  
if b2=120 then Hours2\_Up  
goto Chk\_Day

'minute is up so reset seconds variable  
'increment minutes variable  
'if 2 hour block is up goto Hours2\_Up  
'if not goto Chk\_day to see if is a water day

Station\_1:  
if b2=20 then Station\_2  
let pins=1  
goto Main

'if minutes > 20 goto second station  
'else set pins to turn on Station 1

Station\_2:  
if b2=40 then Station\_3  
let pins=2  
goto Main

'if minutes > 40 goto third station  
'else set pins to turn on Station 2

Station\_3:  
if b2=60 then Station\_4  
let pins=3  
goto Main

'if minutes > 60 goto forth station  
'else set pins to turn on Station 3

Station\_4:  
if b2=80 then Station\_5  
let pins=4  
goto Main

'if minutes > 80 goto Fifth Station  
'else set pins to turn on Station 4

Station\_5:  
if b2=100 then Station\_6  
let pins=5  
goto Main

'if minutes > 100 goto sixth station  
'else set pins to turn on Station 5

'(Last station minute variable value must be less than 120)

Station\_6:  
if b2=119 then Station\_OFF  
let pins=6  
goto Main

'if minutes = 119 go turn off station  
'else set Pins to turn on Station 6

Station\_OFF:  
let pins=0  
goto Main

'last station time up, clear pins to turn off

Hours2\_Up:  
let b2=0  
let b3=b3+1  
if b3=84 then Week\_Up  
goto Main

'last 2 hours is up (1 week), reset minutes  
'increment 2 hour block  
'if week has ended goto Week\_up

Week\_Up:  
let b3=0  
goto Main

'week has ended so reset 2 hour variable