AVR LED Clock

Practice for BASCOM newbies

By **Frank de Leuw**
(Germany)

Anyone who has managed to make an LED turn on and off using BASCOM can also control the segments of a 7-segment display. That is the basis of this simple low-cost digital clock.

When you choose an AVR microcontroller to build a digital clock there really isn't too much extra hardware required. Firmware for the project is written in BASCOM [1], the free demo version of the software development tools is more than adequate for this task. This digital clock design uses a simple software structure which serves as a good introduction to the BASCOM programming language.

One controller, six displays

Hardware for this design's controller board (**Figure 1**) consists of little more than an Atmel ATmega8 microcontroller, an oscillator, two push buttons to set up the time and two pairs of LEDs to provide the colon separators between the displayed time units.

Display multiplexing is performed using two controller ports; port C selects one of the six displays while port D selects one of the seven segments.

The six 7-segment displays are mounted on a separate board and connected to the controller board via six 10-way (2 x 5) connectors (K1 through K6). The display board circuit can be seen in **Figure 2**. The connector also provides

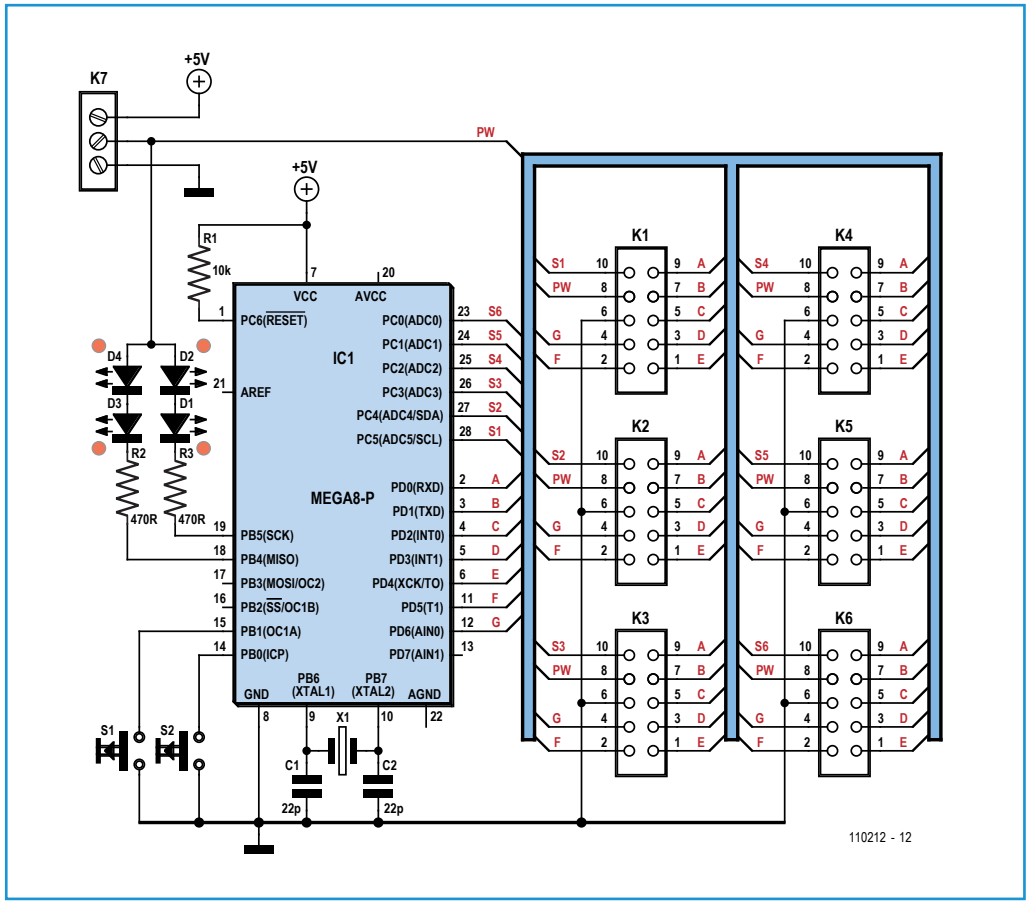

Figure 1.
The clock circuit.

the supply to the display unit. Each display segment is made up of three series-connected LEDs. Current to all the seven segments is a little too high to be sourced from the controller port so transistors T1 and T2 provide a buffer stage. This configuration makes the display more flexible, enabling it to be quite easily powered from an alternative supply voltage. The display supply is connected to pin 2 of the terminal block at K7.

The firmware

In the BASCOM source code for this project [2] [3] the controller type and crystal frequency are first defined and then the ports configured. The two timers are used to generate a one second timebase and also to handle the display multiplexing.

Six variables (*Sec_unit*, *Sec_ten*, *Min_unit*, *Min_tens*, *Hr_unit*, *Hr_tens*) are used together with Timer1 (1 Hz) to calculate the time of day.
Like much software the main program consists of a Do-loop which in this case polls the push buttons and adjusts the displayed time. As far as the main program goes that is about it! When a push button press is detected a subroutine is called which increments the variables *Min_unit* and *Hr_unit*.

The values in Timer0 und Timer1 both generate an interrupt on overflow. The interrupt service routine for Timer1 is called once per second and increments the current time. This routine also takes care of the two blinking colons between hours and minutes and minutes and seconds. The routine is also called when adjustments to the clock time are made.

Timer0 is used to multiplex the display and produces an interrupt at a 400 Hz rate. Each of the six display characters therefore get refreshed at a 67 Hz rate. The If_Then statement assigns a value to each segment and Select_Case directs the value to the corresponding port.

This design has been built many times; it was part of a student course on Mechatronics. It successfully gave them a deeper understanding and confidence working with both the hardware and software aspects of the project. It also provides an ideal opportunity to experiment with BASCOM. If you don't already own a chip programmer there are many programming adapter designs on the Internet which allow you to program the controller chip from a PC. Alternatively you can order a preprogrammed microcontroller from the Elektor shop [3]. The BASCOM source code for this clock is also available to download from the same Elektor project page.

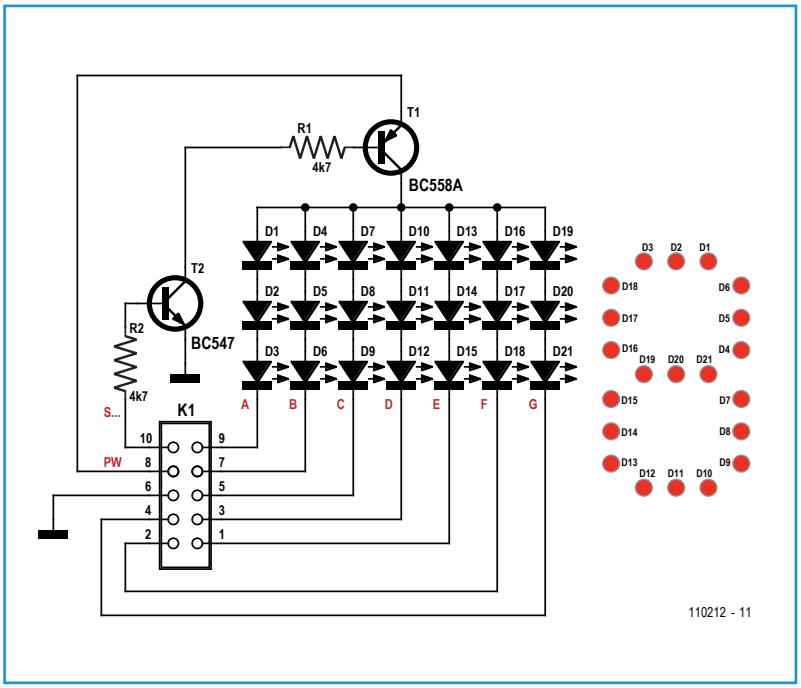

Figure 2.
One 7-segment display unit.

The display is separate from the controller board; this gives you the freedom to be more inventive in the way it's presented. It can of course more conventionally be assembled on a square of breadboard. We have used three LEDs per segment but why not try more? Don't forget the four colon LEDs. Pictures of the prototype design can be found at [2].

(110212)

Internet Links

[1] www.mcselec.com
[2] www.elektronik-radio.de
[3] www.elektor-labs.com/ElektorPost/2013/04