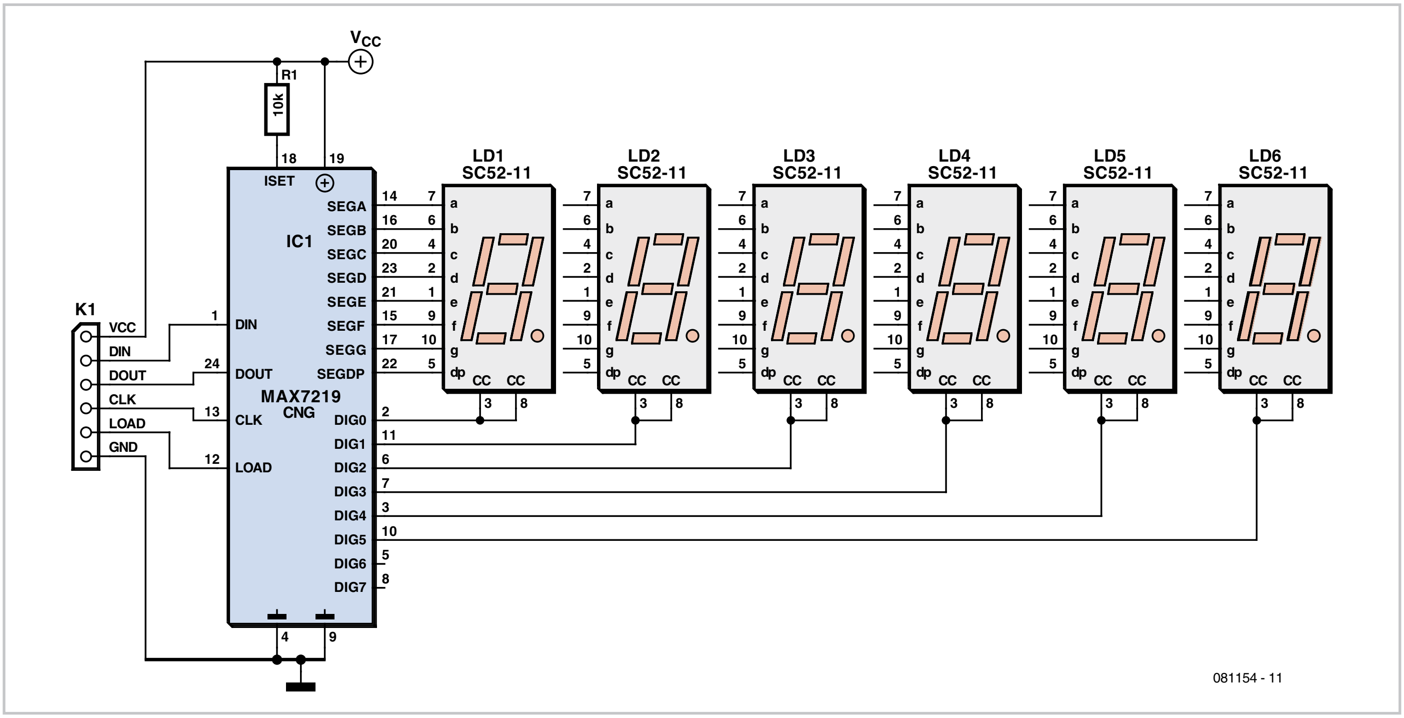
Six-digit Display with SPI Port



Characteristics

- six-digit seven-segment display
- just two components, plus display modules
- driven using software SPI emulation
- C driver routines easily adapted to any type of microcontroller

Uwe Altenburg (Germany)

There is no essential difference between a seven-segment display and seven individual LEDs with either their cathodes or their anodes connected together. Often the display will be driven by a microcontroller, and when several digits are wanted, they are usually driven in a multiplexed fashion. This involves connecting together each segment in a given position across the digits, with each of the seven common segment lines (plus decimal point) being driven by an output port pin of the microcontroller via a series resistor. Each digit also requires a transistor, which again needs an output port pin to drive it. For a six-digit display, this means a total of fourteen output port pins: almost two whole ports on an 8-bit microcontroller.

Maxim offers a solution to this problem in the MAX7219. The device is controlled over an SPI port, requiring just four I/O pins on the microcontroller. It can drive up to eight individual seven-segment displays. Contrary to popular belief, multiplexing the displays does not reduce overall power consumption: although each digit is only driven briefly the LED current must be correspondingly increased in order to achieve the same average brightness. According to the device's datasheet the MAX7219 can deliver up to 500 mA per digit. The rapidly changing current draw can cause interference to the microcontroller's power supply if adequate decoupling is not provided.

An advantage of the MAX7219 is that neither series resistors nor drive transistors are required. Only one external resistor is needed, which is used to set the segment current for all the digits. Since it is also possible to adjust the segment current over the SPI port, a fixed 10 kΩ resistor is suitable.

The small printed circuit board is designed to accept Kingbright SC52-11 common cathode display modules, which have a digit height of 13.2 mm. The display is available in a range of colours. If you wish to modify the board layout to suit a different display, the Eagle file is available for download from the web pages for this article [1].

A special feature of the MAX7219 is the ability to cascade multiple devices, allowing several display boards to be driven from a single microcontroller. No extra I/O pins are required on the microcontroller as the data bits are shifted through the chain of devices: the output DOUT of one module is connected to the DIN input of the next, and the LOAD and CLK signals are wired in parallel.

How do we program the device? The MAX7219 contains 16 internal registers that can be serially addressed and written to. Each seven-segment display is configured using a separate 16 bit message, where bits 0 to 7 carry the data to be displayed and bits 8 to 11 carry the register address. Bits 12 to 15 are not used.

Each bit is clocked into the device on the rising edge of the CLK signal. While the data bits are being transmitted the LOAD signal must remain low; when it goes high the message is written to the addressed register. It is not necessary for the microcontroller to have dedicated SPI hardware: a low data rate is adequate in almost all cases and so the necessary waveforms can be generated in software. The author has written suitable routines in C [1], which are easily adapted for any type of microcontroller. The routine SendCmd() is responsible for 'bit banging' the I/O ports to generate the SPI signals.

A couple of the MAX7219's registers require initialisation. The mode register determines whether the internal BCD-to-seven-segment decoder is used or whether the data stored in the registers correspond directly to segment patterns. The latter option is more general but requires the use of a look-up table in the driver: in the author's source code this array is called Segments. A further register sets the total number of digits to be driven; and finally the segment current must be set and the display enabled. Once everything is initialised the digit registers can be written to using the function UpdateDisplay().

The display module is also supported by the M16C TinyBrick [2] described in the March 2009 issue of Elektor. A simple example program can be downloaded from the project website, showing how easy it is to control the display using the built-in BASIC interpreter.

(081154-I)

Internet Links

[1] www.elektor.com/081154

[2] www.elektor.com/080719

Downloads

081154-1: PCB layout (.pdf), from www.elektor.com/081154

081154-11: source code, from www.elektor.com/081154

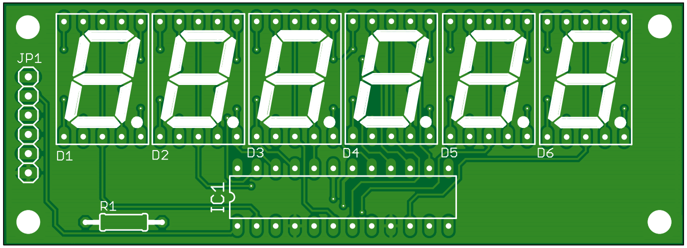
CAD files, from www.elektor.com/081154

COMPONENT LIST

Resistors
R1 = 10kΩ

Semiconductors
D1–D6 = SC52-11 (Kingbright)
IC1 = MAX7219CNG

Miscellaneous
JP1 = 6-way pinheader
PCB # 081154-1 [1]


JP1
D1
D2
D3
D4
D5
D6
R1
IC1