Arduino - 7 Segment or Alphanumeric Displays

Autor:
Data de publicació: 16-12-2013

Tiger 320 Series meters provide the users with static display text messaging. Display text can be easliy edited to suit your application. Scrolling display text messaging requires a simple compiler generated macro.

14-segment alphanumeric display options provide greater legibility than 7-segment options. For example, a setpoint could be edited to read for "tank level low" as:

7-Segment Range:

DI-50, DI-60, DI-503, DI-50AN6, DI-50B51, GI-50, GI-50B51, FI-B101D50

14-Segment Anzeigen:

DI-60A, DI-60AT5C, DI-602A, DI-602AT5C

Download PDF Â» (240.0 KB)

Arduino and TM1638 LED Display Modules
Posted on 11 March 2012. Tags: 7-segment, arduino, chained, daisy, deal, demonstration, display, dx, extreme, green, LED, lesson, lessons, library, red, segment, seven, tm1638, tutorial, tutorials

Introduction

The purpose of this article is to demonstrate the use of some interesting LED display modules I discovered on the dealextreme website, for example:

They contain eight 7-segment red LED digits, eight red/green LEDs and also eight buttons for user input. You can get red or green display models. The units can also be daisy-chained, allowing up to five at once, and a short cable is included with each module, as well as some short spacers and bolts, such as:

The spaces are just long enough to raise the PCB above a surface, however to mount the boards anywhere useful you would need longer ones. You may also want to remove the IDC sockets if you want to mount the module close to the surface of a panel. This would be a simple desoldering task as they are through-hole sockets:

The board is controlled by a TM1638 IC:

This part seems to be a domestic Chinese product from “Titan Micro Electronics“. After a quick search the TM1638 isn’t available from Digikey, Mouser or the element14 group… so if anyone has a lead on a low-volume, reliable supplier for these – please leave a comment below. However here is a link to the data sheet – thanks Marc!.

Getting Started

Now to make things happen…

Hardware – Connection to an Arduino-compatible board (or other MCU) is quite simple. The pinouts are shown on the rear of the PCB, and match the fitting on the ribbon cable. If you look at the end of the cable as such:

The top-right hole is pin one, with the top-left being pin two, the bottom-right pin nine and bottom-left pin ten. Therefore the pinouts are:

Vcc (5V)
GND
CLK
DIO
STB1
STB2
STB3
STB4
STB5
not connected

For Arduino use, pins 1~4 are the minimum necessary to use one module. Each additional module will require another digital pin connected to STB2, STB3, etc. More on this later. Please note that each module set to full brightness with every LED on consumes 127mA, so it would be wise to use external power with more than one module and other connections with Arduino boards. After spending some time with the module, I made a quick shield with an IDC header to make connection somewhat easier:

Software – download and install the T1638 library from here. Thanks and kudos to rjbatista at gmail dot com for the library. Initialising modules in the sketch is simple. Include the library with:

```
#include <TM1638.h>
```