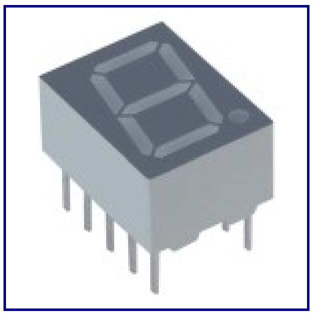
EAM Slave Seven Segment Displays

2010-02-24 16:02:37 by Chris

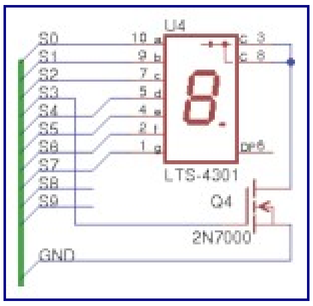


My original plan was for the EAM slave module to have an LCD or VFD screen for the passcode digit display. I'm now in favor of using individual seven segment LED displays instead for the following reasons:

1. Numeric (7 segment) LED displays will be much easier to power than a VFD.
2. LEDs have a broad range in operating temperature (-35°C to +85°C for the unit to the right).
3. Separate displays, compared to one large display, will allow for more flexibility in the layout of the device's front panel. I would rather space 10 digits properly than be forced to use a bulky 20×2 character LCD/VFD screen.

I plan on using the Lite-On LTS-4301JS (see Fig. 1) [Datasheet]. It is a yellow, Common Cathode, seven segment display. They are $0.57 from Digi-Key in quantities of ten (inexpensive compared to similar items).

The disadvantage of driving ten individual 7 segment displays is that each LED must be individually controlled. I searched for an inexpensive display driver (or drivers) capable of controlling ten digits, but could not find a suitable match. Most of the drivers available were made to control only four or eight digits, nearly $10, and surface mount. For this reason I have chosen to drive the LEDs through the microcontroller directly.



I was at first overwhelmed with the idea of controlling 70 LEDs. My initial scheme would use up 17 precious I/O pins ((1 per cathode x 10 displays) + (1 per LED x 7 segments)). This is undesirable for obvious reasons.

Another option is to use a method called Charlieplexing. I found articles on Charlieplexing (1, 2) and devised this scheme which only uses 10 I/O pins. This method involves lots of microcontroller activity however, and I'm not sure that controlling 70 LEDs in this way is the best option. See Fig. 2 for my incomplete schematic.

A third option that I believe I will end up selecting is to use a shift register to control the individual segments and a demultiplexer to address each of the displays. Although this method would involve two additional ICs, it won't bog down the microcontroller and would only use about as many I/O pins as charlieplexing. I don't have a schematic yet for this scheme. As soon as I do I will post it.