

Jeroen Oldenhof

Tech and other fun stuff



OLED display for Alpha JUNO 2

🕒 February 5, 2016 📁 Synths

In my endeavor to replace and upgrade my synths failing displays with more modern counterparts I decided to replace the 1x16 character LCD, EL backlight (and whining inverter) with a nice white OLED display.

I purchased the display through Comet, a european (Bulgarian to be precise) company. It is claimed the OLED display is HD44780 compatible, on which the original Sharp display is based.. and as most of the character LCD's are. For reference, see the datasheet of the original Sharp LM16155 display, and the new Winstar OLED display. Here a sheet with additional data for the OLED controller.



I desoldered the flatcable connector from the original display and soldered it onto the OLED display.



The contrast pin is unused, and the backlight pins are missing, but apart from this the new display is indeed pin compatible with the original one. But.. as you can see, only the first 8 characters are visible. After some reading and searching I learned about the various addressing modes these displays have.

It seems the original Sharp display is not addressed as a 1 line with 16 character, but as a 2 lines of 8 characters display, being the first line the first 8, and the bottom line the second eight characters. Line 1 starts at address 0h, line 2 at 40h. Below tables from the datasheets show the difference more clearly:

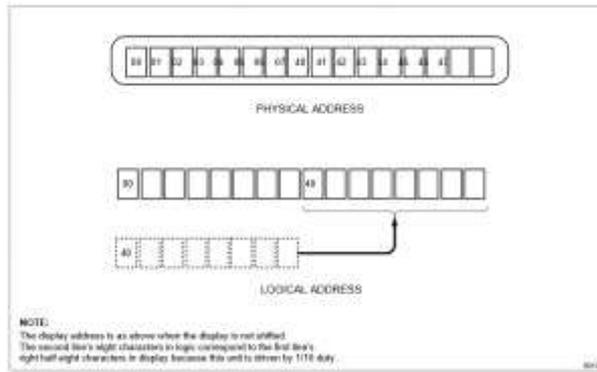
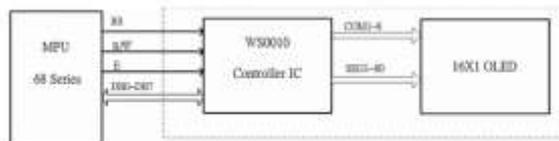


Figure 6. Display Address

HD44780 addressing



Address Format	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
CA (Character Address)	1	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0

1	2	3	4	13	14	15	16
CA=100000000b	CA=100000001b	CA=100000100b	CA=100000011b	CA=100011000b	CA=100011010b	CA=100011100b	CA=100011110b

Winstar OLED addressing

I was hoping the Winstar controller would be able to support this 'older' addressing mode, by setting a jumper.. or something.

I did find the display would accept a 'number of lines displayed' setting in the Function Set setup command. I made a small test-setup using an Arduino controlling the display, so I could experiment with different setup commands and addressing modes. Setting N to 0 or 1 did however not influence the addressing mode. All characters written from address 40h are not visible. No way will this OLED 1x16 display be a direct replacement for a legacy HD44780 based 1x16 LCD display, without altering the way the display is controlled.

N=Number of Lines Displayed – N="1":
2-Line Display – N="0": 1-Line Display

The only way to do this is to hack the Juno's firmware, which is not the road I wanted to go down. I do have some microcontroller assembly programming experience, but there was no source available of the Juno's software, I never disassembled existing binaries like this, and also I did not own an eeprom programmer.

I did however find a binary available from a guy who performs firmware hacks on the JUNO and other synths. After fiddling around with 8051 (which is the Juno's CPU) tools like dis51, d52 and the as31 assembler I was able to create a readable asm file, and assemble it back to a bin file which was the same as the original bin. This gave me confidence to give it a try.

I purchased a nice cheap Chinese EEPROM programmer (thanks to Dave's excellent review on EEVblog), and a couple of EEPROM's of ebay. The original EPROM is a 128kbit NEC D27128D. The EEPROM's I purchased were Winbond W27E257, which are pin compatible with the NEC's. Being EEPROM's, they're easy to reprogram, which I expected to be doing a lot in the process of testing changes made in the software. The Winbond's have a capacity of 256kbit, which were the smallest I could find being pin-compatible and in DIP-28-pin package. The software then has to be uploaded in the upper 128kbit of the Winbond EEPROM, from 4000h, to be addressable for the Juno's CPU.

I spend a lot of time finding the pieces of code where data was written to the display. Below is a subroutine which was easy to find, and also easy to alter. Setting address 0 (first character on first line) is writing 80h to the display, address 40h (first character, second line equivalent to 9th character on 1x16 display) is writing 0Ch. Cutout from the Winstar sheet:

Instruction	Code										Description	Max. Execution Time when fsp or fosc = 250KHz
	RS	R/WB	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Set DDRAM Address	0	0	1	ADD	Sets DDRAM Address. The DDRAM data is sent and received after this setting.	TBD						

In stead of jumping the 40h in addressing we only jump 8h.

```

writeLCD_16_chars_17BC:
    MOV A, #80h
    ACALL writeLCD_8chars_DPTR_17AD
    ACALL delay_3_R0to0_17A6
;    MOV A, #0C0h          ; Go to second row of 8 characters
    MOV A, #88h
    ACALL writeLCD_8chars_DPTR_17AD
    ACALL delay_3_R0to0_17A6
    RET

```

Unfortunately there are a LOT of pieces of code scattered all over the place which all write data to the display, and the jumps of 40h are not always very obvious. To make things even more complicated, there is some shifting going on: When pressing the 'Write' button the existing text is 'shifted' away and new text is written on addresses 88h and 0C8h respectively. Text will shift 'home' when the Write button is released, and original text will re-appear. To make this work with the new OLED addressing this shifting code had to be altered so it doesn't shift two lines by 8, but 1 by 16.

After a couple of nights of tinkering I've gotten all display output correct. I'm not going to bore you with more details, but feel free to drop me a line if you're interested.. or wish to view my annotated assembly code. See attached to final .bin file. I can burn the EEPROM for you if you want.

In the the final step I desoldered the electronics from the PSU board used to power the EL backlight.
No more whining, and the result is simply beautiful!



One remark: the output of dis51 won't assemble straight away in as31, it requires some syntax fixup. I hacked dis51 so it does produce direct usable output, and made it include the pointer address in the labels it produces. I can share this if you want.

3 thoughts on “OLED display for Alpha JUNO 2”



February 6, 2016 at 1:40 am

Looks marvelous. Inspiring work! (You could probably do some business selling kits or doing mods...)

Ultimate
Outsider



February 6, 2016 at 7:04 pm

Hey, great work! I'd love to do this mod as well – possible to buy a programmed EEPROM from you and get a backup of the .bin file as well? If so email me dbalatero@gmail.com

David
Balatero



February 6, 2016 at 9:26 pm

Hi! No problem, I uploaded the the binary (.hex) file to the files section of the 'Vintage synth MOD and repair' facebook group (<https://www.facebook.com/groups/vintagesynthrepair/files/>). I can send you a pre-burned EEPROM as well. Does EUR 40,- (coverage for postage, materials and other expenses) sound ok to you? My paypal address is: jeroen@cj2.nl
Let me know!

★ admin