XPORT and LCD Interface

The XPORT and LCD Interface add-on card occupies one of either Slot1 or Slot2. It performs three basic functions:

- Parallel interface to a 2-line x 24-character LCD display. The 8255-based interface provides 8-bit data I/O via Port A (output and input), and 3-bit control via Port C (output only). The remaining 5 bits of Port C are unused.

- 8-bit LED display. The LED display uses Port B (output only) to control the LEDs. This can be used to replace the HEX LED on the Stack mainboard. In fact, when the EEPROM code is running, the LEDs are used to repeat the HEX LED display.

- Provides power and serial I/O interface lines to an XPORT AR mounted on the adapter card. Although it doesn't happen often, it is possible that the XPORT AR may require that the firmware be reprogrammed. In the worst case, the XPORT AR will be completely out of communication via the LAN-based connection, making it impossible to restore the firmware or otherwise correct the situation. When this happens, serial recovery is the only possible corrective action. (The XPORT will, at all times, retain the 'stub' program that allows for serial-based firmware recovery.) The parallel adapter board provides the serial path needed for the Serial Recovery process.

Parts List

Sockets
2 - 20-pin @ U1, U2
1 - 40-pin @ U3

ICs
1 - SN75185 RS-232 driver/receiver @ U1
1 - 16V8 GAL (pre-programmed TestAdap.PDS) @ U2
1 - 8255 Parallel port @ U3

Resistors
2 - 100 Ohm trimmer potentiometers @ R1, R2
2 - 10 Ohm @ R3, R4
1 - 220 Ohm 10-pin bussed @ RN1

Capacitors
3 - .1ufd bypass @ C1, C2, C3
2 - 47ufd filters @ C10, C11

LEDs
8 - T- 1 3/4 Red or Yellow LEDs @ D1-D8

Misc
1 - XPORT/LCD PCB
1 - Rear Panel Bracket (9-pin D-Sub cutout)
1 - 16" LCD ribbon cable w/16-pin IDC connectors
1 - Serial DB-9 Male connector w/2 Screws
1 - 60-pin (2x30) right angle pin header @ Slot1
2 - 8-pin (1x8) socket headers @ J1, J2
1 - 16-pin (2x8) IDC right angle boxed header
1 - 2x24 Parallel LCD Display (w/16-pin straight boxed header installed)

Construction

Construction is straightforward with no surprises or unusual techniques needed. Orient the PCB such that the "Stack180" etching is face up, towards the lower left. This is the normal 'upright' position. In all cases where positioning is important, pin 1 is the square hole. Perform the assembly as follows:

- Install and solder the sockets at U1, U2 and U3. Pin 1 is marked with a square pad.

- Install and solder the .1ufd mono capacitors at C1, C2, and C3.

- Install and solder the 10 Ohm resistors at R3, R4, and the 220 Ohm bussed resistor at RN1.

- Install and solder the 100 Ohm potentiometers at R1 and R2.

- Install and solder the 60-pin (2x30) card edge connector.

- Install and solder the 16-pin boxed header. Pay attention to Pin 1 alignment.

- Install the two 47ufd electrolytic caps at C10 and C11. In both cases, the '+' side goes in the square hole, towards the 60-pin card edge connector.

- Install the eight LEDs at D1-D8. In all cases, the flat side of the LED goes in the square hole, towards the 60-pin card edge connector.

- Install the two 8-pin header sockets at J1 and J2. You should use an XPORT module to ensure proper alignment of the two sockets. It would be best to use two sets of 1/2" 4-40 standoffs (not provided) to mount the XPORT to the adapter (in the same manner that the XPORT is mounted to the mainboard.) This will ensure perfect alignment prior to soldering.

- Install and solder the male, right angle D-Sub connector at J3. The mounting tabs do not provide enough holding force to keep the riser stiff against the PCB, so you should solder the tabs in the holes provided. This will require a good bit of heat. Be careful that you don't damage the adapter board.

- Install the two 47ufd electrolytic caps at C10 & C11. The positive lead goes in the square hole.

- Loosely attach the rear panel bracket to the D-sub connector with the 4-40 hardware provided. Test fit the adapter to the mainboard (either Slot1 or Slot2) and then tighten the rear panel bracket hardware. A drop of thread locker will ensure the bracket doesn't come loose.

- Connect the ribbon cable between the adapter and the 24x2 LCD.

- If you have not already done so, you'll need to update your EEPROM to version 2.02.00 (dated 8/31/10) or later.

NOTE: Early adapter boards (Ver A) require a series of modifications to the PCB for the serial port. On the back side of the PCB, under the 9-pin connector, you'll note that four traces tied to the 9-pin connector (one pair per side) have been cut. You'll find four small, 30 gauge wires in your kit; use these to re-arrange the connections such that the two trace pairs are 'swapped.' A scanned image is included at the end of this document.

This completes assembly of the parallel adapter board.

Initial Power On Tests

Install the XPORT and LCD Adapter in either Slot1 or Slot2. Connect the 16-conductor ribbon cable between the adapter board and the LCD display unit.

Power up the Stack180. Enter Setup mode, then select option 'C - Motherboard Parameters', followed by sub-option '1 - Toggle LCD Enable.' This will allow the EEPROM code to make use of the LCD hardware on the parallel adapter board. The LEDs on the adapter board are already enabled. They operate in a manner similar to the HEX display, but are obviously limited to a binary display rather than a HEX character display. The top row of LEDs show the status of bits 7-6-5-4; the bottom row displays bits 3-2-1-0.

When you restart the Stack after writing the EEPROM Code Block, you should see the "LCD Display OK!" initialization line as the first entry under the "Initializing..." line. If you see a Timeout error instead of OK, there is a problem with the CPU accessing the LCD display. If the adapter board is not installed, or the LCD is not connected to the adapter, a Timeout error will result.

You'll need to adjust the two potentiometers on the adapter board for best viewing. R2 - the rightmost adjustment (nearest the side with the ribbon cable) - is used to adjust the backlight brightness; R1 is used to adjust the character contrast. Adjust the two controls to meet your needs.

The LCD module will mount in a 5.25" drive bay with the use of a 3.5" - 5.25" drive adapter, although some modifications to the adapter opening will be required. Be very careful - this is part of the 'face' your Stack180 will show.

EEPROM Operation

The EEPROM program (Ver 2.02 or higher) assumes the XPORT and LCD Display Adapter is not installed. You can enable the LCD display in System Setup, Option C (Motherboard Parameters), Sub-option 1 - Toggle LCD Enable.

Once the EEPROM Write process is complete, the Stack will re-boot and attempt to access the LCD display. If the adapter is working correctly, you'll see the 'LCD Display OK' message. Otherwise, the 'LCD Display Timeout' message will be displayed. You'll also see the timeout message if the adapter is installed but the display is not connected.

Once the LCD Display is enabled, the Stack EEPROM program will output the date/time to LCD Line 2 each second. A limited amount of status information is also displayed on Line 1.

The Adapter board also carries 8 LEDs that are used to repeat the HEX LED display, although in a binary format rather than HEX. Any status code displayed on the HEX LED is also displayed on the 8-bit binary LED display.

The LCD display operates under the interrupt system for the 1-second updates. Because interrupts are running and RAM is being changed as a result, you will see RAM test errors when the LCD is enabled. This will be corrected in a future EEPROM update.

The third section of the adapter, the XPORT Serial I/O section, is not accessible from the Stack180. It's purpose is only to provide a Windows machine with serial access to the XPORT AR.

CP/M Driver Operations

When operating under CP/M, a special driver is needed to handle the 1-second interrupt from the Real Time Clock, as well as the handlers needed to control the LCD. The StackOS has been fitted with the required drivers and utility programs; B/P BIOS will follow in the not-too-distant future.

The Epson 72421 RTC is programmed to interrupt the CPU every second. When the interrupt occurs, program execution is diverted to the interrupt handler that will read the current time/date from the RTC, storing it in memory. The process of reading the time/date also clears the interrupt condition.

When the RTC read is complete, the interrupt handler checks to see if the LCD display is present and accessible (as determined during EEPROM startup.) If not, execution returns to the interrupted program.

If the LCD is present and accessible, the interrupt handler will output the current time/date to LCD Line 2. Execution then returns to the interrupted program. This entire process repeats every second.

The interrupt handlers and LCD character output functions are built-in to two different 'programs:' the BIOS and the Extended BIOS. To minimize the BIOS memory footprint, the BIOS RTC interrupt handler merely redirects execution to the External BIOS. An entire 32K bank of Banked RAM is reserved for the External BIOS, so there is plenty of room for future expansion.

During startup, CP/M will load and execute the TBLBANK1.COM program. This loads the External BIOS, provides links to the main BIOS, and enables the RTC 1-second interrupt. (If the RTC 1-second interrupt were enabled during CBOOT, there would not be a handler in place to deal with the interrupt.)

You can compile the BIOS and External BIOS with or without the interrupt and LCD I/O drivers. File A4:HARDWARE.LIB contains two equates - RtcInts and LcdPort - that are used during BIOS and External BIOS compile-time to add or remove the drivers.

Finally, the LCD.COM utility is used to output text to LCD Line 1. The A8:LCD.Z80 file is used to build the program with up to 10 predefined strings. For instance, a command line of LCD 5 will output the predefined string 5 to LCD Line 1. Valid string numbers range from 0 - 9. On delivery, only String 0 is defined - you can define and re-compile the strings however you see fit. Additionally, string 0 is a special case. It not only has a predefined string, it also tacks on a final, 7-byte addition to the string: the hardware that was used to boot into CP/M. For instance, a Primary GIDE boot will display "StackOS Boot on PriGide" when the LCD 0 command is entered.

The last option available to the LCD.COM program is to append a command tail that starts with something other than the numbers 0-9. For instance, a command line of "LCD This is a Test" will display "This is a Test" on LCD Line 1. ZEX batch files can expand the possibilities even further, changing the LCD display based on the status of the ZEX processor.

There is no LCD.COM function for clearing Line 1; instead, the LCD 0 function was intended to return the display to the 'standard' display.

XPORT Serial Recovery

Serial Recovery is used when there is no other way to access the XPORT. Although this is a rare occurrence, when it does occur, it completely removes the XPORT's ability to serve as a gateway for the Stack180 terminal function.

Normally, you can access the XPORT programming/control/status interface via three methods:

1 - Command Line Interface via Stack180 Local Serial port terminal. If the Stack180 is running on the serial terminal, you can 'attach' the XPORT Command Line Interface to the terminal and manually configure each setting. This is done by setting mainboard switch SW1-4 to ON. This setting overrides SW1-1 and forces the terminal on the local serial port, and puts the XPORT into the Command Line Interface mode on power up. This can only be done during the power up sequence - a hard reset will not suffice.

2 - HTTP interface via the DeviceInstaller and the LAN that connects to the XPORT on Port 80.

3 - Command Line Interface via the LAN to XPORT Port 23. Any terminal program that can access an IP:PORT address can be used.

Note that options 2 & 3 can be used even while the Stack is operating on the LAN-based terminal via the XPORT.

The most common situation requiring serial recovery is due to a failed XPORT firmware upgrade. The Stack180 User Manual describes one method that is known to work correctly. Other methods are possible and have led to the need for a serial recovery method, hence the XPORT and LCD Interface Adapter.

To perform Serial Recovery, perform the following:

1 - Shut down the Stack. If your Stack is running on the LAN terminal, disconnect the LAN cable and leave it disconnected. Remove the XPORT from the mainboard mount and install it on the XPORT Adapter that is in Slot1 or Slot2. The 9-pin male serial port on the Adapter is used to connect to your Windows or Linux machine running Device Installer. The port is identical to the serial terminal port, so you can use the same terminal cable if desired. You will not need to be concerned with any terminal I/O - all action will be on the DeviceInstaller program.

NOTE: The serial cable between the two machines is a standard 'null modem' cable, as described in the Stack180 User Manual, Appendix E.

2 - With the XPORT Adapter serial port connected to your Windows/Linux machine, start the DeviceInstaller program. Pull down the Tools → Advanced menu and select 'Recover Firmware.' Select the appropriate COM port. Set the Device Model to XPORT AR. For the Firmware File, select the browse function and find the file labeled "xport_ar_5_1_0_0_R13.romz". This is the latest XPORT firmware file, and is included on the Stack CDROM in the "\Stack Design\Design Files\XPORT AR\FW 5100R13" subdirectory. You may want to completely re-write the XPORT filesystem; if so, check that box.

3 - DeviceInstaller will remind you that the XPORT must be powered off. Click OK, and it will enter a 'wait' status. Start up the Stack. Don't be concerned about any terminal I/O from the Stack - it will be completely ignored during this process. DeviceInstaller will note that the XPORT started up, and will almost immediately start transferring the loader file, followed by the firmware files. The entire process will take about 5 minutes. If DeviceInstaller does not immediately begin downloading the loader file, something went wrong. Check your cable and the connections.

XPORT Serial Recovery (cont)

4 - Once the firmware install is complete, you must configure the XPORT for Stack180 operations. Connect the Stack's LAN cable to the XPORT installed in the XPORT Adapter. On DeviceInstaller, click the 'Search' button and wait for the search to complete.

An XPORT with a 'default' configuration should be found by DeviceInstaller. If the XPORT is not found, you may need to power cycle the XPORT. Before finishing the configuration, you should verify that the XPORT still contains all of the web files within the filesystem. To do so, use DeviceInstaller to drill down through the XPort → XPort AR - firmware v5.1 → (your XPORT IP address.) Click on the IP address, them find the Web Configuration tab in the main window. Click it. If you get a User/password request, enter admin for the user and PASS for the password. Case is important.

If you get a web page that shows the Device Status, then the filesystem is intact and you will not need to re-load it - skip down to the Stack Configuration section below.

Filesystem Reload

With firmware version 5 and up, XPORT configuration pages are no longer loaded via the filesystem. Instead, they are part of the basic firmware download, so a Filesystem reload will not be required.

Stack Configuration

The Stack-specific configuration required for the XPORT will enable LAN-based terminal tunneling, auto power on/off, IP addresses, and will set the baud rates and handshaking required for 230K baud operation. The XPORT uses an XML configuration language that allows you to reload the entire system configuration from a single XML file. You can also save the current configuration into a new XML file if desired.

1 - Start up the Stack. As before, we don't care about any actual terminal I/O. DeviceInstaller will be taking directly to the XPORT - at this point, the Stack is merely a way to provide power to the XPORT.

2 - Start DeviceInstaller and, as before, drill down to the IP of the XPORT. Click on the Web Configuration tab, then the green right arrow over on the right side, and, if prompted, enter admin for user and PASS for password.

3 - In the orange highlighted area, scroll down to the XML tab. Click it, and at the top of the page, select the "Import Configuration" option. Select the "Configuration from External File option, then click 'Browse.'

4 - Use the filesystem browser to locate the file "\Stack Design\Design Files\XPORT AR\xml Stack export.XML." Click on it, then click on 'Import'. Click yes - you really do want to proceed.

5 - Once you receive the line: "All XML configuration records have been imported" you can click on the orange 'System' tab. Click on Reboot.

XPORT Serial Recovery (cont)

6 - After about 30 seconds, DeviceInstaller will attempt to re-open the previously open device. It will probably fail, since the XPORT may be assigned a different IP address. You'll have to click on the 'Search' button.

7 - As before, drill down to the IP address. Open the Web configuration, entering admin for username and PASS for password. Click on the orange 'Status' tab and page down to the Line 1 entry. You'll see the entry is: RS232, 230400, 8, N, 1, Hardware. Line 2 is similar, but at 115,200 baud.

The only configuration items remaining are the IP address and DHCP status. Your XPORT has been reset to DHCP Enabled status as part of the firmware reload, so the IP address was supplied by the DHCP Server in your LAN, making it easy to connect. This type of Dynamic addressing is very easy to set up and use, but usage has shown that a static configuration is usually better for connecting to the Stack180. With a static connection, the IP address will never change and you'll always be able to locate the Stack at the specified address.

If you prefer to stay with dynamic addressing, leave DHCP enabled and exit DeviceInstaller - you're done with the firmware reload.

To set Static addressing, select the orange 'Network' tab, then select the 'Configuration' option. The current selections are from the DHCP server - you should leave most of them as is, changing only as described here:

BOOTP	off
DHCP	off
IP Address:	(See discussion below)
Default Gateway:	as is.
Hostname:	as is
Domain:	as is
DHCP Client ID:	STACK180 (in TEXT mode)
Primary DNS:	as is
Secondary DNS:	as is

Static IP Address

The IP address will have to follow the DHCP-supplied address fairly close. For instance, if the DHCP-supplied address is 192.168.1.xxx, the first three groups of numbers (octets) will need to remain the same as they are. Your only option for setting the static address is in the final octet. Depending on the size of your LAN, this may be pretty easy. A large LAN with multiple access nodes will be a little more difficult.

Usually, your DHCP server will be set to supply IP addresses starting with a specific address for the final octet, such as 50 or 100. In this case, the DHCP-supplied IP address for the XPORT may have been 192.168.1.100 or something similar. In a case like this, you'll normally be safe in assigning an IP address of 192.168.1.200 or higher. (The highest final octet you can use is 255.) You'll have to be certain that no other device is assigned that address. Normally, DHCP prevents conflicts, but setting a static IP address removes DHCP from the equation.

Errata

This photo shows the modifications needed for early Adapter boards (prior to Rev B) in the area of the 9-pin serial connector. Both wire pairs effectively ‘cross’ to re-route the connections.