AST/RO Technical Memorandum #36
15 Jan, 1999

Upgrading Lynx Realtime Operating System on AST/RO Control Computer

Maohai Huang
Department of Astronomy
725 Commonwealth Ave.
Boston, MA 02215

Abstract

This memo describes the upgrading process of LynxOS on the AST/RO control computer (`royal.spole.gov`) to version 2.5.1 from version 2.4.0. Standard installation, system customizing, and driver installation procedures are outlined. Peculiarities during installation and driver debugging are discussed. How problems were solved or circumvented could help future OS upgrading and OS-level system debugging.

1 Introduction

LynxOS is a POSIX compliant realtime operating system that closely resembles a BSD flavored UNIX from a user's view point. Over the years it has evolved from a buggy platform that provided a fragile process execution space and a limited, primitive development environment, to a fairly stable and mature OS that supports a variety of development facilities as well as most UNIX utilities, largely thanks to the sweeping adoption of GNU softwares in LynxOS.

Usually new releases of the OS have known bug fixes and provides the system enhancement that makes the system more robust and versatile, which are the reasons to upgrade the OS. However it is important to realize the cost of upgradings. New OSes usually have their own new bugs. New futures of the new OS often not only are useless for AST/RO but also brings fatal bugs that hinder the use of basic functions of the OS. The disk and memory space occupied by new OS usually grows larger and larger (the "Kitchen Sink"). If the installer does not carefully tailor out some useless default components, a lot of system resources will be wasted on OS components that are not to be used.

Due to custom drivers (AMP, GPIB, and DSP) used by OBS to run AST/RO, upgrading the operating system isn't a straight forward process.

This memorandum first describes the standard upgrading process which involves saving the essential files, extracting the new OS from distribution media, and configuring the system according to the need and limitation AST/RO. Then this memorandum shows the steps to install the custom drivers (as of this writing, they include the AMP, GPIB and DSP drivers) that do not come with the standard release of the operating system.

2 Installing and Configuring the Standard Release of LynxOS 2.5.1

2.1 Save the Essential Files

There are some system files that have the configuration information of the system being used. They should be saved and used as reference to modify the correspondent files after the new OS is extracted from the distribution media. The Lynx Installation Guide has a list of files for a generic OS. This is a list of the files that were saved as a `tar` file in `/usr1` before upgrading to 2.5.1. The directory `/usr1` is a user file system that shouldn't be affected to this OS upgrading.

```
.OBSMACR          .OBSMACROS        .OBSMACROS~
.cshrc            .emacs            .env
.env~             .kermrc           .login
.netrc            .profile          .profile~
.rhosts           .rhosts~          .terms
.xinitrc          OBSMACROS_5_19_98.save OBSMACROS_8_25.save
/sys/devices/ampinfo2.c /sys/devices/dspinfo.c
/sys/drivers/amp/ /sys/lynx.os/CONFIG.TBL
/etc/passwd       /etc/tty
/etc/exports      /etc/hosts         /etc/tconfig
/etc/starttab     /net/rc.network    /bin/ksh
```

To be safe, another copy of the `tar` file is saved on a different machine, although saving it on a QIC tape is highly recommended, because if for some reason the new OS doesn't have the network working properly after it is installed, and `/usr1` is damaged, a `tar` file residing on a different machine is difficult to copy over to the new OS.

It is urged to make a full system back up before a upgrading. The back up is very useful or vital in case the user file system are corrupted or destroyed, or when, after the new OS has erased the old OS, the installer wants to read what a certain system file that have not been saved, or if the installer simply wants to switch back the old operating system.

In the above files besides the usual UNIX system files (`.profile`, `passwd` etc.) there are OBS-relation files like the `.OBSMACROS` and a driver directory `/sys/drivers/amp`. Also there is `/bin/ksh` which is a Korn Shell. The `/bin/ksh` has small footprint and is likely more compatible to existing shell scripts than the Bourn-Again Shell (`/bin/bash`) that is the default of LynxOS shell when the OS is shipped from the factory. LynxOS used to have a very buggy /bin/sh written in-house. To circumvent it AST/RO has been using the Korn Shell that was compiled in the BL. Lynx has switched to GNU's Bourn-Again Shell as the default, and discarded their own version of `sh`. Some existing AST/RO shell scripts still use `/bin/ksh`. Currently there seems no need to switch to `/bin/bash`.

2.2 Extract the New OS Release from Distribution Media

Follow the instruction of the Lynx Installation Guide to boot the computer with the bootable disks and run the installation script. The installation script will ask questions regarding the system configuration of the new system.

Because there is no CDROM reader on the control computer, a QIC tape has to be made from the CDROM `/tar_images` directory using the `dd` command, per instruction of the Guide. On the QIC there should be one `tar`-format file that can be un-`tar`'ed by the `tar` command. It should be examined bye a `tar tvf /dev/rsd1542.2` command on royal before the real upgrading.

Here are some information of the system that could help the installer to make decision on what to answer (Information here based on system configuration as of this writing) to the installation script:

- The destination device to install the new OS is `/dev/sd1542.0a` (`1542` – SCSI card type 1542, `0`– ID#0 on SCSI interface, `a` – first partition).

- The source device is `/dev/rst1542.2` (That means raw SCSI interface type 1542, SCSI ID #2)
- The Ethernet interface is wd3e
- YES to install TCP/IP.
- The host name is royal and the IP is 204.89.132.132
- YES to install both NFS server AND client
- Leave STREAM interface enabled.
- YES to virtual memory. The swap file name is `/.swap` and the size is 32 (MB)

After the installation begins, the new OS will un-`tar` from the tape.

Take the floppy disk out so that when extraction is finished, the installation will start the secondary installation process (rebuild new kernel etc.) and boot the machine to the hard disk, on the new OS.

2.3 Configure the System to Restore the Configurations Saved before the Upgrade

At its end, the installation script will attempt to reboot the machine to the new OS.

There is one catch in the AST/RO system at this stage. The interrupt vector of the ethernet card is set to 5 in the default kernel, while the ethernet card is actually configured to have interrupt number as 10. Because of this, the machine will hang up if the new OS is booted to multiuser mode (it starts TCP/IP automatically in multiuser mode). To change the interrupt vector in the kernel, one needs to edit the file `/sys/devices/if_wd3einfo.c`, so that the line "`32 + 5`" becomes "`32 + 10`". Then type "`make`" in the

directory, followed by `cd /sys/lynx.os; make install`. Since the machine will hang if booted up by default, in order to do the kernel re-making, there are several ways to boot the system up without starting the network. One is booting the system to single user mode when the preboot program prints out "`Command? <b s0a /lynxos>`". Another is press control-C right after the device `/dev/sd1542.0a` is mounted on `/` but before TCP/IP initialization is started (this might need some practice. Reading `/bin/rc`, one can see /net/rc.network is run after the disk mounting and date setting).

After the kernel is re-made and the machine is booted to multiuser mode using the new kernel, all the network utilities should work. If not, the configurations in `/net/rc.network`, particularly the parameters of `ifconfig` and `route` may not be set right. The file of the same name in the backup should be consulted to modify the default `rc.network`. In the same fashion all files that have been backed up should be looked at and correspondent file in the new OS should be modified accordingly. Note that often it is not a good idea to over-write the default new system files with the ones in the `tar` file, since there are sometimes changes in details of these file when new version of OSes come out. An understanding of those system files is important. Most of these files have man pages that explain the format and meaning of the contents of the files.

At last, note that, the time zone setting and virtual memory on/off are in `/bin/rc` file. A link `/bin/sh` should be made to `/bin/ksh` which is extracted from the `tar` file.

After these changes, the system should be able to boot to a usable OS that is close to what it had been like before the upgrading for a normal user.

2.4 Some Fine-tuning of the OS

For AST/RO the `vdt`, `sim1742`, `sim2742`, `sim2940`, `scsi810`, `hds`, and `ptr` drivers can be commented out from `/sys/lynx.os/CONFIG.TBL`. The file `fd.cfg`, `sim1542.cfg` in `/sys/cfg` can be modified so that only the first two devices are kept remain in the system. Although `hd` (IDE disks) are not used in AST/RO system, commenting them out in `CONFIG.TBL` will result the major device numbers of the SCSI1542 disks to change, which makes the next rebooting tricky. So it's better to leave `hd.cfg` line alone in the `CONFIG.TBL` file. See the comment in `CONFIG.TBL` for more information. The above changes can be skipped if size of the kernel isn't an issue.

After commenting these not-to-be-used drivers out, remake a kernel by typing "`make install`" in `/sys/lynx.os` and "`reboot -aN`".

Note that several `simXXXX.cfg` files in `/sys/cfg` has the line "`n:ratape:@rstXXXX.2::`" in them. They serve to make symbolic link "`ratape`" pointed to the `rstXXXX.2` device. This link is for Lynx `tar` to use as the default tape device. However, the default `CONFIG.TBL` has all the `simXXXX.cfg` included. Since it's the last "`n:ratape:@rstXXXX.2::`" line that determines which device `/dev/ratape` points to, if `sim1542.cfg` isn't the last SCSI disk configuration in `CONFIG.TBL`, the link will be made pointing to a device that doesn't exist in the system. So the installer should either comment out every `simXXXX.cfg` line after the 1542, or, in the `CONFIG.TBL` file, add a line "`n:ratape:@rst1542.2::`" at the file end.

2.5 Additionals

X and Motif is usually not used for AST/RO so they are currently not installed. Also the RAM capacity of the computer royal is being too small for running X.

MASTERLYNX is distributed in a separate tar file in the CDROM. The "micro http daemon" `mhttpd`, if is needed, has to be extracted separately.

3 Installing Device Drivers that are used by AST/RO

AST/RO uses several device drivers that are not distributed in the standard release of Lynx Development OS. The installation and debugging of these drivers are discussed here. Because drivers work inside the kernel, it is often difficult to debug them if something goes wrong.

Some of the drivers used by AST/RO was developed long before Lynx became matural. The way these drivers were written may cause errors when they are compiled, linked to the kernel, installed (during booting process), or used. In particular, there are code written in assembler language in the Intel style. Lynx's old buggy ("Legacy") C compiler should be used to compile these code. However, now Lynx has adopted the GNU C Compiler (`gcc`) as the default compiler for applications and the kernel. The `gcc` uses Motorola style for in-line assembler code, so the old style assembler code need to be compiled separately. Lynx has renamed the old `cc` to `/bin/l-cc`. A change in the `makefile` would have been sufficient to compile the assembler source files using `/bin/l-cc` with some exceptions (see the GPIB section below). It is likely that Lynx will eventually phase out the `l-cc` and force users to re-write their Intel assembler code.

Here we discuss three drivers: the AMP driver for the Arnet RS232 Multi-port, the GPIB driver for the NI ESP488 board, and the DSP driver that provide timing to OBS.

3.1 The AMP Driver

The Arnet Multi-Port driver has been around since circa LynxOS v2.0 in 1992. It provides eight RS232 serial port devices (`/dev/amp[0-7]`) for the OS. This driver has been working reliably, although there have been changes made to it so that it would work for "new and improved" versions of OSes.

The first change involves the `sgtty` structure. Starting from LynxOS2.2 `<termio.h>` was included in the source code because of system header file changes of the OS.

When the control computer of AST/RO ran out of serial ports in 1998, a DIGI Classic board is installed in the system to emulate the then out-of-production Arnet board. It provides additional eight RS232 lines. The AMP driver is adapted to create devices `/dev/dig[0-7]` that uses the same AMP driver at different IO addresses and different interrupt number. The changes are made in `ampinfo.c` and `amp.cfg` (now named `amoinfo2.c` and `amp2.cfg`).

To install the drivers, cd to `/dev/drivers/amp` then

- copy `amp2.cfg` to `/sys/cfg`
- copy `ampinfo2.c` to `/sys/devices`
- copy `ampinfo.h` to `/sys/hdeaders`
- edit `/sys/devices/Makefile` to add `ampinfo2.x` at the end of the line that starts with `DEVICE_FILES_all`, and at the end of the file, add a line "`ampinfo2.o:  $(DHEADERS)/ampinfo.h`"
- in `/sys/devices` type `make install`. this for one thing will add `ampinfo2.o` to the library file `/sys/lib/devices.a`
- type `make ampdrvr.o`. This is the driver file

- make a directory `/sys/xdr`, and copy `ampdrvr.o` to `/sys/xdr`. For some obsecure reason `ampdrvr.o` cannot be added to the library file `/sys/lib/drivers.a` and be linked to the kernel correctly. So my work-around is that copy `ampdrvr.o` to `/sys/xdr`, and
- modify `/sys/lynx.os/Makefile` to add `/sys/xvd/*.o` to the end of line that starts from "`COSTUMER_DRIVERS = `". So when the new kernel is made, `ampdrve.o` will be linked to the new kernel.
- add a line "`I:amp2.cfg`" to the end of `/sys/CONFIG.TBL`
- type "`make install`" in `/sys/lynx.os`. This will have the new kernel made and installed to `/`, among other things.
- take a look at `/etc/nodetab`. You should see lines like `amp0:c:??:0:0666` in it.
- type `reboot -aN` to reboot the system

When the system comes up again, connect a terminal to one of the serial ports and use kermit to test if the driver works. Be sure to set the baud rate correctly on the terminal and kermit sides. The configuration is often: parity none, 8 bits, no parity check, stop bits 1. Edit `/etc/ttys` to set up login line and default baud rate (ref man pages of `ttys` and `tconfig`).

3.2 The GPIB Driver

A more complicated driver is the ESP GPIB driver to interface the OS to the National Instrument GPIB master card.

The driver source code and testing programs are provided by NI. The user is supposed to install the driver on the OS and use library calls defined in `cib488.c`.

To install the driver, cd to `/sys/drivers/gpib/source`,

- copy `ibinfo.c` to `/sys/devices`
- copy `cib488.h` to `/sys/dheaders`
- copy `ib.cfg` to `/sys/cfg`
- edit `/sys/devices/Makefile` to add "`ibinfo.x`" at the end of the line that starts with "`DEVICE_FILES_all`", and at the end of the file, add a line "`ibinfo.o:  $(DHEADERS)/cib488.h`"
- in `/sys/devices` type `make install`
- make `esp488.o` and `ibio.o` (see notes below)
- copy `esp488.o ibio.o` to `/sys/xdr` (make sure the directory exists, and that `/sys/lynx.os/Makefile` have been modified. Ref. AMP driver installation guide above)
- add a line "`I:ib.cfg`" to the end of "`/sys/lynx.os/CONFIG.TBL`
- add `/lib/libc.a` to the end of "`OTHER`" line in `/sys/lynx.os/Makefile` It defines `_Binsrch` which somehow is referenced in GPIB driver code.
- make install in `/sys/lynx.os` and `reboot -aN` the system.

Note that because of a `l-cc` bug (and the lack of commitment of Lynx to fix anything in `l-cc`), `esp488.c` cannot be compiled entirely using `l-cc`. The binary object file produced by `l-cc` is corrupted and will hang up the OS if used. So the assembler code in `esp488.c` (and files included therein) are put in a separate file called `ibio.c`, and is compiled with `l-cc`. The rest of `esp488.c` is compiled with `gcc`. This has caused changes in `esp488_master.c`, `sys488_master.c`, as well as the `Makefile`.

Testing programs `ibtsta` and `ibtstb` can be used to test the driver after the system is booted to the newly made kernel.

Since the testing utilities don't come with source code, a demo program called `tg.c` can be compiled and used to write something to the HP Multmeter. This serves as a straight forward test of the GPIB software and hardware.

3.3 The DSP Driver

More than one DSP drivers have been developed for OBS. The current one is used due to its simplicity and reliability. The installation is from the directory `/usr1/obs/src/drivers/dsp`. The installation wan't done in a similar way to how the AMP and the GPIB drivers were installed, particularly on the device installation part.

In directory `/usr1/obs/src/drivers/dsp`,

- make new `dspinfo.o` and `dspdrvr.o`
- copy the above files to `/sys/xdr`
- copy `dsp.cfg` to `/sys/cfg`
- add a line "`I:dsp.cfg`" at the end of file `/sys/lynx.os/CONFIG.TBL`
- make install in `/sys/lynx.os`
- Reboot the system: `reboot -aN`

To see if it works, run OBS and `namon` to see if the tracker can update time. The program `/usr1/obs/src/drivers/dsp/test/tst.c` can be used to test if the DSP driver is producing 10ms timing. Make sure the first DSP card is loaded with program `aosmon6.mpo`. Program `loadt.c` can be used to load this program. Run OBS will usually load, too.

3.4 Debugging the Drivers

The functions provided by Lynx OS `kprintf()`, `kkprintf()`, and `cprintf()` can be used to print out debugging information inside driver code. LynxOS has moved `kkprintf()` to a undocumented place `/sys/lib/kernel.a`. This file will have to be included during the linking of the new kernel if `kprintf()` or `kkprintf()` is used.

Often the system is needed to be rebooted many times when new kernels are made for debugging purposes. Since bad kernels often hangs the system completely, it is often necessary to press the *RESET* button on the computer chasis to reboot the system. A lengthy fsck will ensue. So it is convenient to copy all the driver code to somewhere in file system `/` (`/sys/drivers` is a good place), and don't mount `/usr1` and `/data` files systems while debugging the drivers.

The Lynx technical support email adress is `support@lynx.com`.

Lynx has a web address at `http://www.lynx.com`.