Motion Engineering

33 South La
Patera Lane
Santa Barbara, CA 93117-3214
ph (805) 681-3300
fax (805) 681-3311
tech@motioneng.com

Release Note
MPI/XMP-Linux
Support Package

MPI/XMP-Linux Library Version: 20030620.1.11
XMP Firmware Version: 495A7

Revised 12Feb2004
DCR 745

1. Introduction

Welcome to the MPI/XMP-Linux Support package. The distribution was built and tested with Red Hat Linux 8.0, kernel version 2.4.18. The libraries have been built with standard GNU tools included with the Red Hat software. This document provides an overview of the release and describes the new features and changes from the standard WinNT MPI/XMP software releases.

Each MPI/XMP-Linux distribution has a particular revision date. To properly run client/server applications between Linux and WinNT systems, the WinNT systems must have the standard, WinNT MPI/XMP release of the identical revision date.

	New Version	Previous Version
Firmware	495A7	459B3
MPI/XMP Library	20030620.1.11	20030620.1.8
MPI/Linux Library	20030620.1.11	20030620.1.8

1.1   System Requirements

This release requires an Intel x86 processor, running Linux Red Hat 8.0, kernel 2.4.18 software.   Only use the standard GNU tools that are supplied with the Red Hat 8.0 distribution.

Target System:	x86 platforms
Linux Distribution:	RedHat 8.0
Kernel Version:	2.4.18
Supported Compiler:	GNU C Compiler Version

1.2   Installing the Distribution on the Host System

The MPI/XMP-Linux installation consists of an TAR file base distribution. The recommended installation directory is /MEI. To install the compressed release, you need to do the following as the 'root' user:

```
mkdir /MEI
mv XMP_Series0620.1.1_Linux.tar.ZIP /MEI/
cd /MEI
tar xvfp *.tar
```

The release will require a few minutes to extract to disk.

The MPI/XMP-Linux installation is designed as a stand alone MPI/XMP release.
All Linux specific files are installed into Linux/x86 subdirectories. Key components of the distribution are:
  • Linux driver, installed in the **(/MEI)/XMP/MPI/Linux/x86/driver** subdirectory.
  • The release and debug versions of the MPI/XMP-Linux libraries, installed in the **(/MEI)/XMP/ lib/ Linux/x86** subdirectory.
  • GNU Makefiles for various sample applications and utilities.

2. General Operation

2.1 Configuration Utility

The map file must be specified with the Configuration Utility by using the *-map* option.

```
config -map 495A7.map -file filename
```

2.2 Starting and Building the MEI/XMP driver

The MEIXMP driver needs to be loaded each time the system is rebooted. A generic x86 driver for Linux is included in every release in the **(/MEI)/XMP/MPI/Linux/x86/driver/Release** directory. This driver was built using Red Hat Linux 8.0, kernel 2.4.18. A debug version of the driver is located in the (/MEI)/XMP/MPI/ Linux/x86/driver/Debug directory. Motion Engineering recommends using the release version of the Linux driver.

To load this driver, you will need to do the following as the 'root' user:

> $ cd /MEI/XMP/MPI/Linux/x86/driver/Release
> $ insmod meixmp.o (loads driver)
> $ grep meixmp /proc/devices (returns device "major_no")
> $ mknod /dev/xmpcontrol0 c "major_no" 0 (makes boot device)

To build and load a custom driver for your Linux system:

> $ cd /MEI/XMP/MPI/Linux/x86
> $ cd driver/Release
> $ make -f meixmp.mak
> $ insmod meixmp.o (loads driver)
> $ grep meixmp /proc/devices (returns device "major_no")
> $ mknod /dev/xmpcontrol0 c "major_no" 0 (makes boot device)

2.3 Host/Client Communication between WinNT and Linux Applications

The target system is accessible from a WinNT client through TCP/IP via the *server.c* utility. The *server.c* source code is included in the directory **(/MEI)/XMP/util/**. A compiled version of the utility is provided in **(/ MEI)/XMP/bin/Linux/x86**. This process must be spawned to receive any commands from a WinNT host. Press <ESC> to begin the process shutdown of the server.c thread. To completely exit the application when running under Linux, press <Ctrl> + C. This is different than WinNT systems, which only require <ESC> to exit Server.exe. On the Client side, all WinNT utilities and sample programs must be run with the *-server 'target'* flag.

2.4   Running XMP support utilities under Linux

Pre-compiled utilities are included in the Linux release. These utilities support two modes of communication - they can be loaded and run from the target system or across ethernet from a WinNT host with the *-server* flag.

NOTE: Linux is case sensitive.

Here is the syntax for a few of the support utilities:

flash **program**
- From the **(/MEI)/XMP/bin/Linux/** directory, type:
        ./flash ../../XMPnnnXn.bin
- Running via TCP/IP from a WinNT host, type:
        flash -server 'target' XMPnnnXn.bin

VM3 **program**
- Operation not supported on the Linux system.
- Running via TCP/IP from a WinNT host, type:
        VM3 -server 'target'

Motion Console **program**
- Operation not supported on the Linux system.
- Running via TCP/IP from a WinNT host. For operation instructions, please refer to the Motion Console section under the Utilties menu on MEI's Technical Support website (http://support.motioneng.com/).

Motion Scope **program**
- Operation not supported on the target system.
- Running via TCP/IP from a WinNT host. For operation instructions, please refer to the Motion Scope section under the Utilties menu on MEI's Technical Support website (http://support.motioneng.com/).

2.5   MPI/XMP Sample Applications

This release installs a *makefile* for building the sample applications with GCC. The *makefile* provides an interface to build all the sample apps from the command line using the *make* utility.

3. Known Bugs and Issues

There are no known bugs or open issues in the Linux support package. Please see the general release notes for outstanding MPI/XMP bugs and limitations.

3.1 Warning Messages

The following warning message is returned when loading the driver:

> Warning: loading meixmp.o will taint the kernel: no license
> See http://www.tux.org/lkml/#export-tainted for information about tainted modules
> Module meixmp loaded, with warnings

This message will appear because the driver has not yet been registered. This warning does not affect the operation of the driver in any way and should be ignored.

4. Limitations

4.1 Using the Server Utility with Multiple Processes (Applications)

A limitation exists with the Linux operating system, which prevents two applications (processes) from safely running at the same time. This limitation is a result of the way semaphores and mutexs are used. In Linux, mutexs and semaphores only lock across threads operating in the same process, which breaks the process-safe design of the MPI. Therefore, you cannot run two applications safely at the same time on the target machine.   Until Linux supports locking across processes with semaphores and mutexs, there are two workable solutions.

The recommended solution is to run all applications through the server utility. This will allow all applications to access the controller through the same process. Please refer to the Server Utility documentation on MEI's Technical Support website (http://support.motioneng.com) for more information.

The alternative solution is to spawn the server utility in a separate thread within your application, so that the application and the server utility are in the same process. Source code for the server utility can be found in **server.c**, which is shipped with the standard MPI software package. Separate applications can access the motion controller through this same server process.