

Advanced Realtime Control Systems, Inc.

ARCS LIGHTNING SOFTWARE INSTALLATION SUPPLEMENT FOR LINUX VERSION 1.2

4 March, 2001 http://www.arcsinc.com

© Copyright 2001 by Advanced Realtime Control Systems, Inc. All Rights Reserved

Contents

1	Introduction	1
2	Software Components To be Installed 2.1 Common Initial Installation Steps 2.2 ARCSware Installation 2.3 JRE Installation 2.4 C Compiler and Run-time Library Installation 2.5 Linux Driver Installation 2.5.1 How to Start the Linux Driver 2.5.2 Setting Defaults for Boot-up of Hardware Host 2.6 Common Closing Installation Steps	$f 1 \\ 2 \\ 3 \\ 3 \\ 3 \\ 4 \\ 4 \\ 4 \\ 5 \\ 5 \\ 1 \\ 1 \\ 2 \\ 1 \\ 2 \\ 1 \\ 2 \\ 1 \\ 2 \\ 1 \\ 2 \\ 1 \\ 2 \\ 2$
3	Configuring for ARCS Users	5
4	Starting ARCS	6
5	Sample Project	6
6	Linker File Notes	7
7	Uninstall	8

1 Introduction

This document supplements the ARCS Lightning Hardware and Software Installation Manual. It provides instructions for installing the ARCS base product software on a linux operating system.

It is strongly recommended that you also read beforehand the Overview section in *ARCS Integrated Development Environment (AIDE) User's Manual*, even if you do not plan to use *AIDE*. That section provides a description of the ARCS control system, its capabilities, and key terminology.

2 Software Components To be Installed

There are two ARCS software components to be installed on a linux system:

- linux driver. To be installed on the Hardware Host only.
- ARCS ware.

The Hardware Host computer i.e., the computer that houses the ARCS Lightning DSP Board, requires both components. A remote computer that will be a client only requires ARCS ware.

Additionally:

- Java Run-time Environment (JRE) v1.1.8 v3 is required on every computer that will perform any ARCS functionality.
- Gnu C4X C compiler and Run-time Library is required on any computer that will be used to develop applications.

To perform installation, follow the instructions in:

- 1. Sec. 2.1, initial installation steps.
- 2. Each subsection which applies to the component you wish to install on this computer (e.g., *ARCS*ware, linux driver, JRE, C4X Compiler and Run-Time Library)
- 3. Sec. 2.6, closing steps.

2.1 Common Initial Installation Steps

This section describes the initial installation steps common to all software components.

- 1. You should have root privileges when you do the install.
- 2. Do one of the following:
 - (a) You can perform the installation directly from the ARCS Linux Distribution CD. In this case, references in this document to <yourTempDir> would mean your CD-ROM drive directory (e.g., /mnt/cdrom).
 - (b) OR you can copy the installation tar file on the ARCS Linux Distribution CD into a temporary directory. In this case, references in this document to <yourTempDir> would mean this temporary directory. Unpack the arcs linux installation file by issuing the following command:
 - tar -zxvf <.tar filename>

You now have a subdirectory called *arcs4linux_1_2b*.

3. Change into arcs4linux_1_2b subdirectory.

 $ເ in this supplement as <ARCS_HOME>$. The default <ARCS_HOME> is */usr/local/arcs*. If you wish to change this default, edit the ARCS_HOME variable in the Makefile in *arcs4linux_1_2b* before continuing installation.

2.2 ARCSware Installation

When you are in the arcs4linux_1_2b directory, type make install.

2.3 JRE Installation

- 1. Change into the directory where you want to install the JRE. A commonly used directory to hold the JRE is */usr/local*.
- 2. Type tar -zxvf <yourTempDir>/jre/<.tar.gz filename> to extract the files.
- 3. One of the files that is extracted is the *jre* application located in jre118_v3/bin. Copy that file into a directory that is in *every* ARCS user's path, such as /usr/local/bin, or put <JAVA_HOME>/bin in the PATH.

2.4 C Compiler and Run-time Library Installation

To install the gnu C4X C Compiler:

- 1. When you are in <yourTempDir>, change into the c4x subdirectory.
- 2. type make. This will install the c4x packages in the required order.
- 3. The c4x files are now installed in /usr/local/c4x directory. The run time library is also installed as part of this process.

2.5 Linux Driver Installation

- 1. When you are in *arcs4linux_1_2b*, type make install-driver.
- 2. Then type make compile-driver to compile the driver to your kernel level.
- 3. When you wish to start the driver, see next section.

2.5.1 How to Start the Linux Driver

To start the driver, reboot the computer. You may also manually start the driver by typing

/sbin/insmod <ARCS_HOME>/driver/arcs01x.o.

To check that the driver started properly, you can use the linux command dmesg to display the status of all devices on your computer, including the ARCS linux driver.

2.5.2 Setting Defaults for Boot-up of Hardware Host

You can configure ARCS to automatically do one or both of the following at Hardware Host Boot-up: Search for the correct I/O base port on the ARCS Lightning DSP Board and start the Hardware Host Server.

The file /etc/sysconfig/arcs contains the configuration settings that can be adjusted as follows:

• I/O Base Port. If *PORT=0* (default setting), then the system will scan the ARCS Lightning DSP Board during boot-up of the Hardware Host to find the correct I/O Base port. Alternatively, you can explicitly set the hex address and thus no scanning is done. For example, to set the address to hex 360 the format is: *PORT=0x360*. Please refer to *ARCS Lightning Hardware and Software Installation Manual* for more information on the hardware port.

Hardware Host Server. If HDWHOSTAPP=0 (default setting), then the Hardware Host Server is not started automatically at boot-up.
If HDWHOSTAPP=1, the Server is started automatically. Please refer to the Overview section in ARCS Integrated Development Environment (AIDE) User's Manual for an explanation of Hardware Host Server.

2.6 Common Closing Installation Steps

After you have installed the software components, do the following:

- 1. Read and follow the steps in Sec. 3, configuring for ARCS users.
- 2. Verify that the appropriate references are made to the JRE directory by doing the following:
 - Change into */usr/local/bin*
 - You will see the *Aide* file, *MonitoringApp* file, and, if you installed the linux driver, the *HdwHostApp* file.
 - Check the JAVA_HOME variable in these files. The default value is JAVA_HOME=/usr/local/jre118_v3. If this is not what you set during installation to hold the JRE, modify the JAVA_HOME variable as appropriate.
- 3. Start the Linux Driver if you installed it on this computer.

3 Configuring for ARCS Users

Every user on this computer who will use ARCS must have their own

arcsware.props file in their home directory. The arcsware.props file specifies initialization values and other information such as the working directory. The working directory is the default root directory for project files, source code, executables, data acquisition files, etc. Each ARCS user on this computer can have their own working directory or the users can share the same working directory.

To make a copy of the arcsware.props file, do the following: (This can be done by the root administrator or the user.)

- 1. Copy the template file called <*ARCS_HOME*>/arcsware.props.orig into the user's home directory and call the file arcsware.props.
- 2. Modify the user's *arcsware.props* file as appropriate (e.g. change the prefix */home/user* to reflect the user's home directory, adjust working directory if desired.).

4 Starting ARCS

- To start *AIDE*, type Aide at the command line. The *AIDE* manual contains details on launching and using *AIDE*.
- To start a Monitoring Client, type MonitoringApp at the command line. You will then be asked to supply the URL of the Hardware Host you wish to monitor.
- To start the Hardware Host Server, type HdwHostApp on the Hardware Host at the command line.

5 Sample Project

Once AIDE is started, you can test the functionality by building a sample project.

- 1. Select the *Project Builder* tab in the *AIDE* window.
- 2. Click on File from the menu bar and select Open.
- Change directory to sample-prj subdirectory of <ARCS_HOME> and select the sample project sinewave_example.prj.

- If <ARCS_HOME> is not /usr/local/arcs, you will need to add the following files to the sample project:
 - (a) source file sinewave-dac.c
 (located in <ARCS_HOME>/sample-prj)
 - (b) header files actuator.h, host_comm_01x.h, hdw_util.h, data_acq.h (all located in <ARCS_HOME>/dsp/inc).
 - (c) ARCS library file libarcs01a_sm.lib(located in <*ARCS_HOME*>/*dsp*).
 - (d) Linker file arcs01.cmd(located in <ARCS_HOME>/dsp).
- 5. Please read the section on linker files and make the appropriate adjustment if applicable to your installation.
- 6. Click on My Projects
- 7. Click on the file name sinewave_example. Press your right mouse button and select Build. An executable file sinewave_example.out will be generated and stored in the executable subdirectory specified in arcsware.props. You can now run this file on the ARCS Lightning DSP Board. For details on using *AIDE*, please refer to the *AIDE* manual.

6 Linker File Notes

- Linker files arcs01.cmd and arcs02.cmd assume that the ARCS Lightning DSP Board has 128K memory. If the board memory size is 512K or 640K, the corresponding link files should be used. For example, for a 640K board, the files arcs01_640K.link and arcs02_640K.link should be linked to arcs01.cmd and arcs02.cmd.
- 2. Linker file arcs02.cmd allocates twice the amount of dynamic memory than arcs01.cmd. Dynamic memory in ARCS ware is used primarily for data acqui-

sition. Hence, a larger dynamic memory allocation would mean that more data points can be saved. However, there would also be less space for the executable program.

3. The user can modify the dynamic memory size directly by changing the parameter __SYSMEM_SIZE in the linker file. This parameter is set to 0x4000 in arcs01.cmd and 0x8000 in arcs02.cmd.

7 Uninstall

If you need to uninstall ARCSware, go to *arcs4linux_1_2b* and type make clean. If you need to remove c4x and jre, you will need to do so manually.

Index

ARCSware Installation tar file, 2 temporary installation directory, 2 uninstall, 8 arcsware.props, 5 C compiler installation, 3 requirement, 1 Configuration for multiple users, 5 Driver configuration, 4

installation, 4 start, 4

Java

JRE installation, 3 requirement, 1

Linker file, 7

Sample project, 6