

Checking Linux Dependencies for CCSv5

Contents

Overview

Checking Dependencies (CCSv5.4 and v5.5)

Generating the List of Required Dependencies

Script Download

Searching for and installing missing packages

Overview

When installing CCS v5.4 and v5.5 on Linux, you may find CCS will not install or will not work because not all the required Linux packages have been installed. The script described here provides a way to determine what dependencies are required and check if these dependencies are installed.

The dependency checker is built into the CCSv6.0 and up installer, so you do not need to run it separately.

Checking Dependencies (CCSv5.4 and v5.5)

To check if you have all the required dependencies, download and unpack the attached TAR file into a local folder. In a terminal window, run the `./check_depends.sh` script. The output will look something like this:

```

Installed versions of glibc and glibcxx: GLIBC_2.15, N/A
Required versions of glibc and glibcxx: GLIBC_2.7, GLIBCXX_3.4.16
Checking availability of libraries in ext-depends.txt ...
Failed to find lib: libhistory.so.4
Failed to find lib: libnotify.so.1
Failed to find lib: libodbcinst.so
Failed to find lib: libodbc.so
Failed to find lib: libreadline.so.4
Done.

```

The main difference between CCSv5.4 and v5.3 w.r.t. dependencies is v5.4 includes a version of `libstdc++` with the install.

Generating the List of Required Dependencies

The script works by reading a list of dependencies from a text file and checking if these libraries can be loaded. If you have a CCS install for which you want to regenerate this dependency list, you can do this by editing the `generate_depends.sh` script to point to the CCS install location and then running the script. After the script completes, re-run the `check_depends.sh` script.

Script Download

http://software-dl.ti.com/dsps/dsps_public_sw/sdo_ccstudio/scratchpad/depends_o.2.tar.gz (http://software-dl.ti.com/dsps/dsps_public_sw/sdo_ccstudio/scratchpad/depends_o.2.tar.gz)

Searching for and installing missing packages

Each Linux family has a different way of searching for and installing packages. The list below explains how to do this for particular distributions. This list is a work in progress and will grow over time. Feel free to add to this list.

(1) Ubuntu

```

## To search for packages containing libjpeg8
# dpkg -l | grep libjpeg8

ii libjpeg8:amd64      8c-2ubuntu7      amd64 Independent JPEG Group's JPEG runtime library (dependency package)
ii libjpeg8:i386       8c-2ubuntu7      i386  Independent JPEG Group's JPEG runtime library (dependency package)

## Alternatively:
# apt-cache search "libjpeg8"
libjpeg-turbo8 - IJG JPEG compliant runtime library.
libjpeg8 - Independent JPEG Group's JPEG runtime library (dependency package)
libjpeg8-dbg - Independent JPEG Group's JPEG runtime library (dependency package)
libjpeg8-dev - Independent JPEG Group's JPEG runtime library (dependency package)

## We want libjpeg8:i386, IE the 32-bit version of libjpeg8. To install it
# apt-get install libjpeg8:i386

```

Change log:

Apr 25/2013: (v0.2) Add an exclusion list so libraries that are not actually used don't appear as requirements.

[Amplifiers & Linear](#)
[Audio](#)
[Broadband RF/IF & Digital Radio](#)
[Clocks & Timers](#)
[Data Converters](#)

[DLP & MEMS](#)
[High-Reliability](#)
[Interface](#)
[Logic](#)
[Power Management](#)

[Processors](#)

- [ARM Processors](#)
- [Digital Signal Processors \(DSP\)](#)
- [Microcontrollers \(MCU\)](#)
- [OMAP Applications Processors](#)

[Switches & Multiplexers](#)
[Temperature Sensors & Control ICs](#)
[Wireless Connectivity](#)

This page was last edited on 11 September 2018, at 14:49.

Content is available under Creative Commons Attribution-ShareAlike unless otherwise noted.