

Linux-c6x-2.0-Beta2 Release Notes

Introduction

This is the 2.0 Beta2 release of Linux-c6x. This release demonstrates Linux on the c66x and c64x+ devices. Users are encouraged to update to later releases when available. Bug fixes will be applied to subsequent releases of the 2.0 series.

Release highlights

- Hard Floating Point ISA Support for c66 (*New for Beta2*)
- Big Endian Support for c66 (*New for Beta2*)
- Re-added Support for c64x+ (*New for Beta2*)
- Add NAND boot support to existing TFTP boot in c64x+ devices (*New for Beta2*)
- Continued Support for c66 little endian
- Based on Linux kernel v2.6.34
- uClibc v0.9.31-rc
- busybox v1.17.1
- Drivers such as i2c, Ethernet, UART.
- Support for peripherals such as EEPROM and NAND
- Supports SysLink IPC (only c66x little endian)
- MCSDK (Multicore SDK) web control panel demo
 - includes Multicore elfloader and IPC demo through SysLink IPC and BIOS/IPC. (only c66x little endian)
- Supported devices
 - EVMC6678
 - EVMC6670
 - EVMC6474L
 - EVMC6474
 - EVMC6472
 - EVMC6457
 - DSK6455

This release uses an updated, early-access version of the gcc tool chain (4.5-124) ^[1] from Code Sourcery.

To build SysLink and the BIOS examples or to program the bootloader, you will need other resources. This release was tested with the versions of these components contained in CCSv5.0.3.00028 ^[2].

This release was built on Ubuntu 10.04 32 bit and that is the recommended host configuration. Other recent 32 bit Linux distributions are expected to work but may require extra Linux knowledge. 64 bit hosts are not expected to work at this time. Older Linux distributions (for example Red Hat Enterprise Linux 4) would require extra work and updated software and is not recommended. Building on non-Linux hosts is not supported at all; use a virtual Linux host if you must use Windows, Mac, etc.

This release uses the CGT based TI tool chain only to compile the BIOS example programs. It is not validated for producing Linux binaries.

Location of the Release Files

The Software can be downloaded from ti.com Software Download Site ^[3] or from Linux-c6x.org File Download Area ^[4].

For this release, you will need to separately download the bootloader and support files. These are only available at the Linux-c6x.org File Download Area ^[4]. In future releases the bootloader and support will be included in the main release files.

Using Prebuilt Binaries

The binaries for your selected platform can be downloaded from the locations mentioned above. Please see the User's Guide ^[5] for instructions on using the prebuilt binaries.

Building Software from the Source

Initial Linux Host Setup

- Please prepare the Linux Host with the appropriate software.
- For Ubuntu 10.04 Host, use the following commands:-

```
sudo apt-get install -y build-essential git-core expect automake
echo "no" | sudo dpkg-reconfigure -f teletype dash
```

- For others, refer [[6]]

If building SysLink and the BIOS examples, you will need to download and install the version of CCS specified above. Download requires filling out a form but is free of charge. The C6000 build time components of CCS are available with the free license as is JTAG debug with all the above supported boards. Use with other boards or emulators may require a paid license.

Cloning the git repositories

Issue the following commands

```
mkdir ~/my-linux-c6x
cd ~/my-linux-c6x
wget http://linux-c6x.org/bootstrap
chmod +x bootstrap
./bootstrap linux-c6x-2.0-beta2
mkdir -p downloads
```

Getting sources from the tar file

As an alternative to cloning from the git repositories, you can download the release tar file and extract it. This may be easier for you if you are behind a firewall but it will only let you build this one release.

You do not need to do this if you have already cloned the git repos.

Issue the following commands

```
wget http://linux-c6x.org/files/releases/linux-c6x-2.0-beta2/
linux-c6x-2.0-beta2-src.tar.gz
tar xvzf linux-c6x-2.0-beta2-src.tar.gz
cd linux-c6x-2.0-beta2
mkdir -p downloads
```

Configuration and Build

```
cd linux-c6x-project
./prj config
[Please edit setenv to specify what to build, or leave it untouched to
build the example configuration]
./prj config
source setenv
make product
```

Additional information

By default the build will generate kernels for the 6678 EVM. The **setenv** can be modified easily to build kernels for other EVMs. The best source of information to figure what to change in the setenv is the setenv itself. It is an excellent README in itself which also has instructions on how to build SysLink

If changes are made to the setenv, please source the setenv for the updates to take effect. Issue the following command before issuing a make product.

```
source setenv
```

If changes are made to the external dependencies specifiers (GCC_VERSION, CCS_VERSION, CCS_DIR, etc) you will need to rerun the configure step. Issue the following commands:

```
./prj config
source setenv
```

Where to find the generated files?

Once the build is successful, the generated files can be found at ~/my-c6x-linux/product. Here it is assumed building only evmc6670.

```
cd ~/my-c6x-linux/product
ls -l

-rwxr-xr-x  1 a0868495 clearntp  4632132 Jul 27 11:14
vmlinux-2.6.34-evmc6670.el-20110727-1
-rwxr-xr-x  1 a0868495 clearntp   3916288 Jul 27 11:14
vmlinux-2.6.34-evmc6670.el-20110727-1.bin
-rwxr-xr-x  1 a0868495 clearntp      7295 Jul 27 11:57 bootblob
-rw-r--r--  1 a0868495 clearntp  3684416 Jul 27 13:43
mcsdk-demo-root-c6x.cpio.gz
```

How to create a bootblob?

The bootblob binary under product directory can be used to generate a bootblob that can be programmed into the nand flash.

To create a bootblob do the following

```
cd ~/my-c6x-linux/product
chmod +x ./bootblob

./bootblob make-image --abs-base=0x80000000 --round=0x100000 blob.bin
vmlinux-2.6.34-evmc6670.el-
```

```
20110603-1.bin mcsdk-demo-root-c6x.cpio.gz "console=ttyS0,115200
mem=256M initrd=0x%fsimage-start-
abs-x%,0x%fsimage-size-x% ip=dhcp rw"
```

The generated file, blob.bin can be used for programming the NAND flash. Refer linux-c6x-2.0-Beta2 User Guide for details on how to program the NAND flash

Memory partition information

On C6670 and C6678, Linux NetCP driver uses a part of the MSMC memory. Similarly SysLink uses DDR and MSMC memory for Shared Region. Here are the memory range used...

User	memory	Range
=====		
NetCP	MSMC	0x0C000000-0x0C003000
SysLink	MSMC	0x0C008000-0x0C0F8000 (For SharedRegion 0)
Linux	DDR	0x80000000-0x8FFFFFFF (Managed by Linux on core 0, assumes mem=256M on command line)
SysLink	DDR	0x9FC00000-0x9FFFFFFF (For SharedRegion 1 and User specified SharedRegion)

Multicore loader application

The mcoreloader program is available to load images on slave cores. The application currently supports statically linked elf images and is demonstrated through Syslink sample applications while loading BIOS IPC ELF image on slave cores. The mcoreloader is included in the mcsdk-demo file system.

Issues Fixed in this release

GDB support for C66x devices

Known issues

- Kernel - Shared EMAC has problem in building due to defconfig file.
- Kernel - The Flash file system build for NAND is not building with mkjffs2 command.
- Kernel - Require example code demonstrating EMAC functionality across multiple cores running heterogeneous OS
- Kernel - kernel clean up pending for usage of cache.h from user space
- LTP - LTP watchdog test fails on c64x and c66x
- SysLink - Syslink/OOB demo would need to handle modification in IPC_ResetVector with updated IPC versions
- SysLink - Assert log message seen when doing rmmod syslink.ko after running messageQ sample application
- SysLink - GateMP, on RTOS side sendEvent fail log message is seen (but sample application runs to completion)
- SysLink - Assert log message seen when running HeapMemMP sample application
- Tools - No Utility to query CPSW statistics
- SysLink - performing a build with SysLink enabled takes a long time
- Bootloader - on the 6670, NAND boot is not functional in big endian

Support

Please send any support related questions to the TI e2e forum ^[7]

References

- [1] <https://support.codesourcery.com/GNUToolchain/release1882>
- [2] http://software-dl.ti.com/dsps/forms/self_cert_export.html?prod_no=setup_CCS_5.0.3.00028.tar.gz&ref_url=http://software-dl.ti.com/dsps/dsps_public_sw/sdo_ccstudio/CCSv5/CCS_5_0_3/
- [3] http://software-dl.ti.com/sdoemb/sdoemb_public_sw/linux_mcsdk/02_00_BETA_2/index_FDS.html
- [4] <http://linux-c6x.org/files/releases/linux-c6x-2.0-beta2/>
- [5] <http://linux-c6x.org/files/releases/linux-c6x-2.0-beta2/linux-c6x-2.0-beta2-users-guide.pdf>
- [6] http://www.linux-c6x.org/wiki/index.php/Setting_up_the_Linux_Host
- [7] <http://e2e.ti.com/support/embedded/t/354.aspx>

Article Sources and Contributors

Linux-c6x-2.0-Beta2 Release Notes *Source:* <http://ap-fpdsp-swapps.dal.design.ti.com/index.php?oldid=111635> *Contributors:* BillMills, Raghunambiath, Spaulraj