# Clock Tree Tool User Manual version 1.7

## Texas Instruments OMAP44xx Multimedia Device™ Family of Products

# Technical Reference Manual

OMAP)))™
TEXAS INSTRUMENTS TECHNOLOGY

TEXAS
INSTRUMENTS
OMAP

**Clock Tree Tool User Manual version 1.7**

# IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

| Products | | Applications | |
|---|---|---|---|
| Amplifiers | amplifier.ti.com | Audio | www.ti.com/audio |
| Data Converters | dataconverter.ti.com | Automotive | www.ti.com/automotive |
| DSP | dsp.ti.com | Broadband | www.ti.com/broadband |
| Clocks and Timers | www.ti.com/clocks | Digital Control | www.ti.com/digitalcontrol |
| Interface | interface.ti.com | Medical | www.ti.com/medical |
| Logic | logic.ti.com | Military | www.ti.com/military |
| Power Mgmt | power.ti.com | Optical Networking | www.ti.com/opticalnetwork |
| Microcontrollers | microcontroller.ti.com | Security | www.ti.com/security |
| RFID | www.ti-rfid.com | Telephony | www.ti.com/telephony |
| RF/IF and ZigBee® Solutions | www.ti.com/lprf | Video & Imaging | www.ti.com/video |
| | | Wireless | www.ti.com/wireless |

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

# EXPORT NOTICE

Recipient agrees to not knowingly export or re-export, directly or indirectly, any product or technical data (as defined by the U.S., EU, and other Export Administration Regulations) including software, or any controlled product restricted by other applicable national regulations, received from Disclosing party under this Agreement, or any direct product of such technology, to any destination to which such export or re-export is restricted or prohibited by U.S. or other applicable laws, without obtaining prior authorisation from U.S. Department of Commerce and other competent Government authorities to the extent required by those laws.

This provision shall survive termination or expiration of this Agreement.

According to our best knowledge of the state and end-use of this product or technology, and in compliance with the export control regulations of dual-use goods in force in the origin and exporting countries, this technology is classified as follows:

- US ECCN: 3E991

- EU ECCN: EAR99

And may require export or re-export license for shipping it in compliance with the applicable regulations of certain countries.

# Trademarks

OMAP, TMS320C55x, C55x, TMS320DMC64x, C64x, SmartReflex, NaviLink, WiLink, BlueLink, M-Shield, FlatLink3G, Code Composer Studio and ICECrusher are trademarks of Texas Instruments Incorporated.

ARM7 ISA, ARM926EJ-S, ETM, Cortex-A8, NEON, and TrustZone are trademarks of ARM Limited, and Jazelle RCT and Thumb-2 are registered trademarks of ARM Limited.

Bluetooth is a registered trademark of Bluetooth SIG, Inc. and is licensed to Texas Instruments.

Mobile CMADS is a trademark or registered trademark of NEC Electronic Corporation in Japan and other countries.

MagicGate and Memory Stick are registered trademarks of Sony Corporation, and Memory Stick PRO is a trademark of Sony Corporation.

HDQ is a trademark of Benchmarq.

1-Wire is a registered trademark of Dallas Semiconductor.

Symbian OS is a trademark of Symbian Software Ltd.

Linux is a trademark of Linus Torvalds.

3GPP is a registered trademark of European Telecommunications Standards Institute.

DiskOnChip is a trademark of M-Systems Flash Disk Pioneers.

Mentor Graphics is a registered trademark and Inventra is a trademark of Mentor Graphics Corporation

Windows is a trademark of Microsoft.

PowerVR is a registered trademark of Imagination Technologies Limited.

Universal Scalable Shader Engine (USSE) is a trademark of Imagination Technologies Limited.


All other trademarks are the property of their respective owners.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1. Clock Tree Tool History

## 1.1. CTT Version History

**Table 1.1. CTT Release Notes**

| Version | Release Date | Notes |
|---|---|---|
| v1.7.0.0b | January, 2010 | • Initial OMAP 4430 CTT beta release version |
| v1.7.0.0 | April, 2010 | • Pin names renamed as follow: pad_slimbus_clks to abe_slimbus1_clock slimbus_clk to slimbus2_clock pad_clks to abe_clksk extalt_clkin to fref_xtal_in sys_clkin to fref_xtal_in xclk_60m_otg to usba0_ulpiphy_clk xclk_60m_hsp1 to usbb1_ulpiphy_clk xclk_60m_hsp2 to usbb2_ulpiphy_clk<br>• PER_ABE_NC_FCLK divider associated register changed from CM_CAM_FDIF_CLKCTRL to CM_SCALE_FCLK[0] SCALE_FCLK<br>• PER_96M_FCLK, EMIF_L4_ICLK and C2C_L4_ICLK clock sources added<br>• SLIMBUS_CLKS and PAD_CLKS switches changed from hardware to software controlled<br>• USB PHY and USB UTMI PHY merged into one module block<br>• CTT Application Report menus and functionality included |
| v1.7.0.1 | July, 2010 | • OMAP4430 ES1.0 to ES2.0 migration<br>• slimbus pin names swapped<br>• timers muxes inputs swpped<br>• INIT_96M_FCLK clock is removed<br>• UNIPRO1 and DPLL_UNIPRO modules removed<br>• DPLL_SYS_REF_CLK divider is removed, DPLL_SYS_REF_CLK becomes equal to SYS_CLK<br>• New multiplexor for ABE_DPLL_BYPASS_CLK is added. ABE_DPLL_BYPASS_CLK can be sourced by SYS_CLK or 32K_CLK using two switches<br>• Move Optional INIT_32K_FCLK clock enable bit for USBPHY module to Always on domain. List of registers impacted: CM_L3INIT_USBPHY_CLKCTRL, CM_ALWON_USBPHY_CLKCTRL<br>• L3INIT_DPLL_ALWON_CLK clock is renamed to USB_DPLL_CLK that is only used by DPLL_USB<br>• PER_DPLL_EMU_CLK clock, sourced by DPLL_PER M7 output, is removed. PER_SGX_FCLK source is updated to DPLL_PER M7 output<br>• Added C2C_L3X2_ICLK, comes from the same node as L3_ICLK, goes to C2C<br>• Added 32K_FCLK, goes to 32KTIMER module<br>• INIT_48MC_FCLK input to HSUSBHOST as mandatory clock. In the same time it is still used as an optional clock, too |

| Version | Release Date | Notes |
|---------|--------------|-------|
|  |  | • MUX_DBG_CLK now switches b/w EMU_SYS_CLK and CORE_DPLL_EMU_CLK when DPLL_CORE is locked |
| v1.7.0.2 | August, 2010 | • Divider_TRACE_CLK and Divider_STM_CLK associated registers are swapped<br>• Divider_MPU_DPLL_HS_CLK and Divider_IVA_DPLL_HS_CLK divider factors changed from 1,2,3,4 to 1,2,4,8<br>• Code Composer Sudio (.gel file) and Lauterbach (.cmm file) scripts included for the register dump functionality (see Section 3.5.4.4) |
| v1.7.0.3 | September, 2010 | • Splash screen fix for Linux not loading properly<br>• SR clock domain source clock changed from WKUP_L4_ICLK2 to L4WKUP_ICLK |
|  |  |  |

# Chapter 2. Clock Tree Tool and JGraph Library Installation

## 2.1. CTT Installation

⚠️ **Caution**

Java™ Runtime Environment, Standard Edition (v 1.6.0 build 17-b04 or higher) **must** be installed before Clock Tree Tool is installed.

The Clock Tree Tool (CTT) installation procedure is composed of following steps.
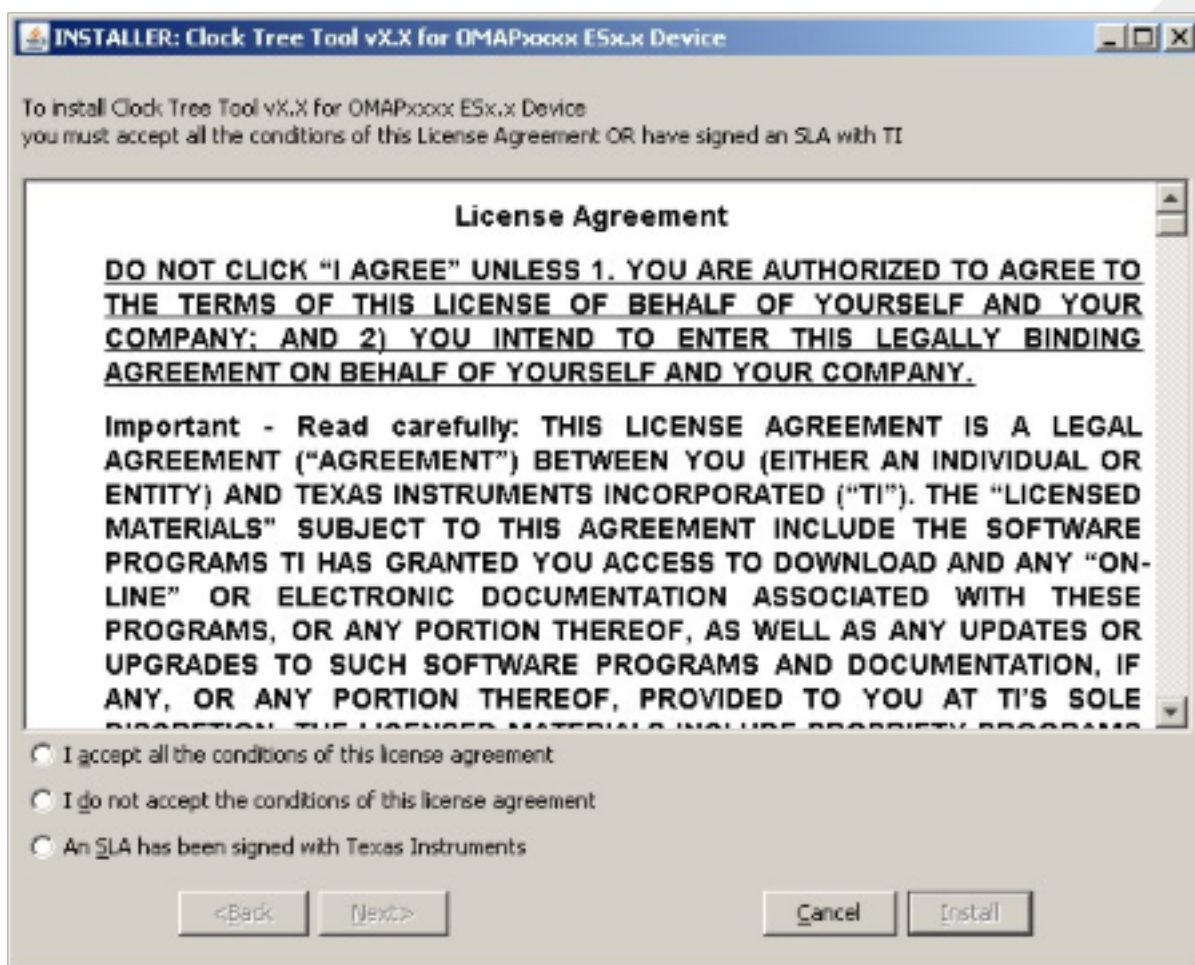
### 2.1.1. CTT Installation Step 1

The CTT installtion procedure gives the end user an option to select the desired installation application. To begin installtion, double click on the "Installer-CTT-OMAP4430ES*x.x*-v*x.x.x.x*" file. A question will appear asking for user selection. The two options are:

- *Install CTT and Application Report*. This is the full installtion package containing the Clock Tree Tool and the Application Report. If selected, please proceed to CTT Installtion Step 2
- *Install Only Application Report*. This is the low level requirement installation. This could be particularly helpful if the user does not have desire or resources to run the full CTT installtion. If selected, a window will be displayed. Select the disired destination and click "Save". The Application Report will be installed in the <Selected Destination Folder>/ TI_Clock_Tree_Tool_APN/<Application Report>. Please skip the rest of the CTT user manul instructions.

### 2.1.2. CTT Installation Step 2

To install the Clock Tree Tool double click (java -jar in terminal for Linux users) on the "Installer-CTT-OMAP*xxxx*ES*x.x*-v*x.x.x.x*" file. The installer will execute and display the License Agreement window shown in Figure 2.1. You must either accept the conditions of the license or have a signed SLA to proceed further. Select an option.

**Figure 2.1. CTT Installation License Agreement Window**



## 2.1.3. CTT Installation Step 3

When the conditions of the license agreement are accepted or the SLA option is selected, the "Next" button will become active. See Figure 2.2. Then, click on the "Next" button to proceed to next step of selecting an install directory followed by the Export Control Notice Window showin in Figure 2.4

**Figure 2.2. CTT Installation License Agreement Window 2**
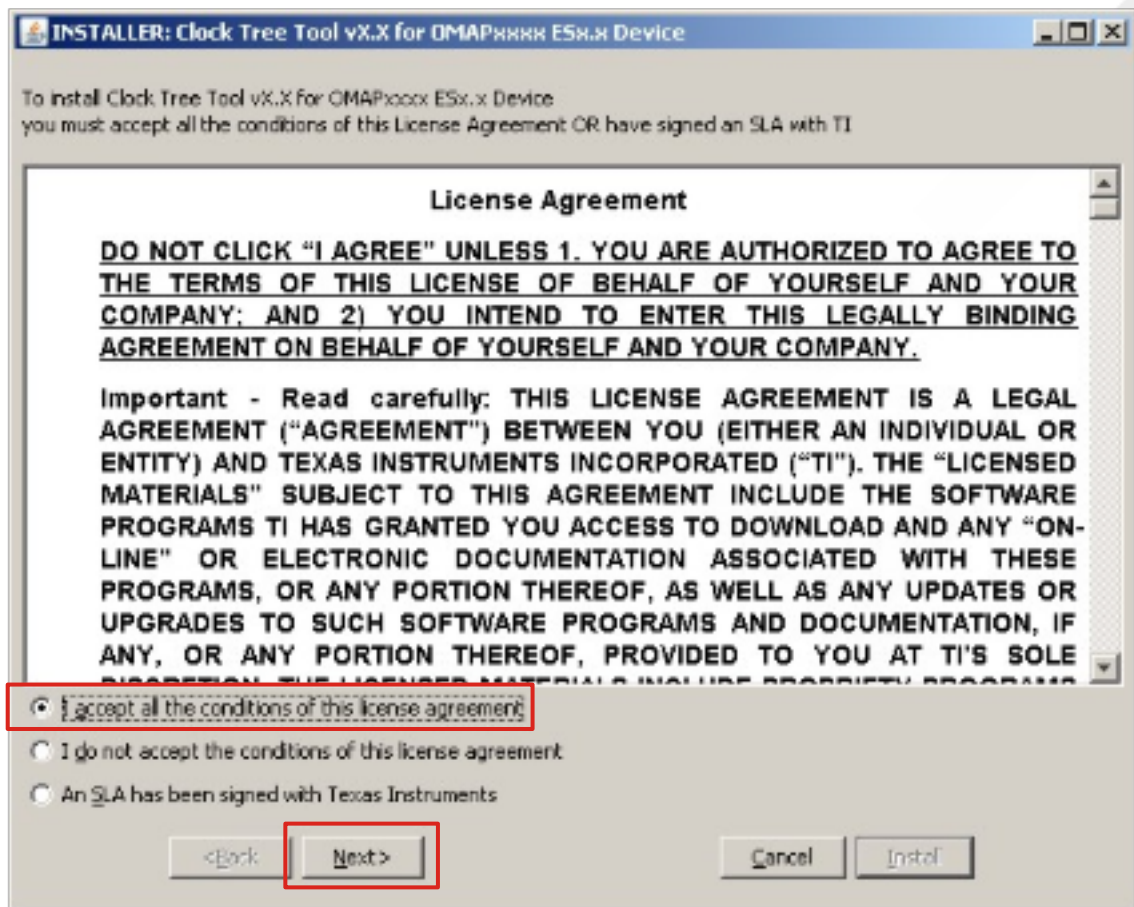


## 2.1.4. CTT Installation Step 4

In the "Export Control Notice" window, the "Install" button is enabled. Click on the "Install" button to proceed to the "Destination Directory Selection" window Figure 2.3 It allows the user to identify the directory for installation of the Clock Tree Tool. Once the directory is selected click the "Select" button to start the installation.

**Figure 2.3. CTT Installation Destination Directory Selection Window**
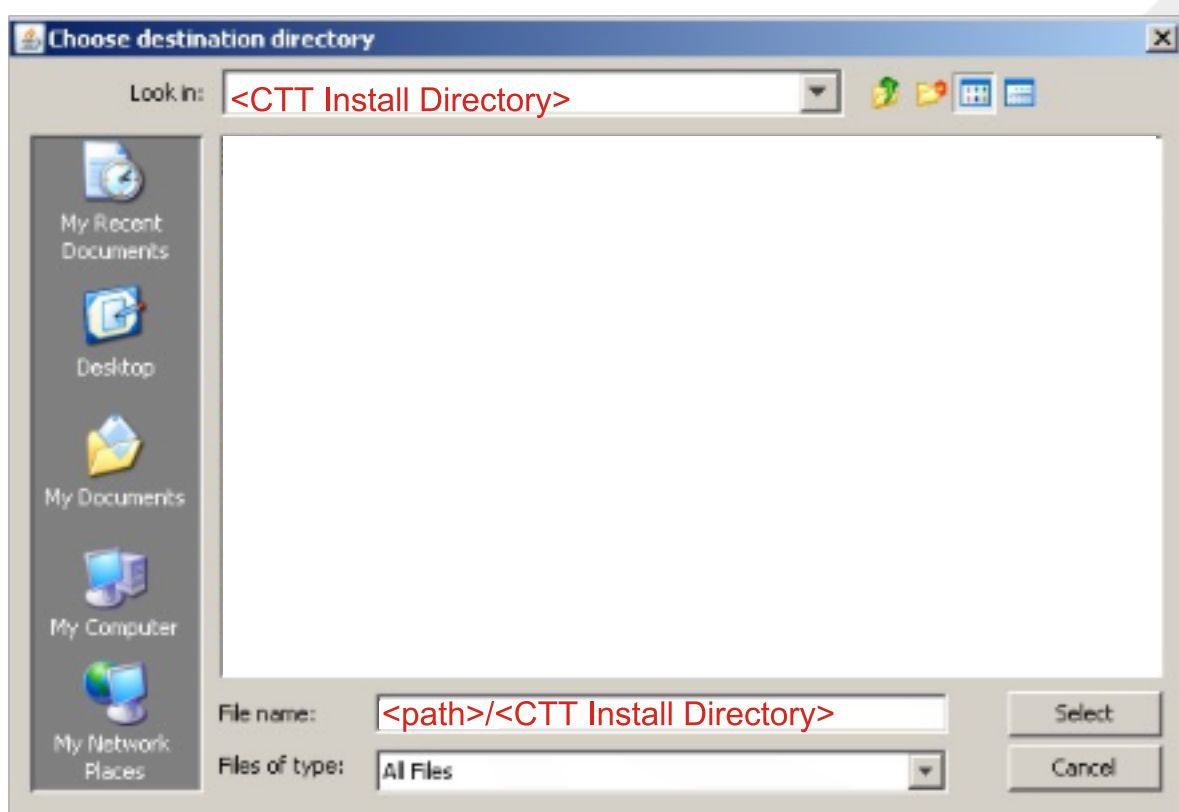


## 2.1.5. CTT Installation Step 5

When the installation is complete the "Installation completed" message is displayed Figure 2.4 Click on "OK" button to proceed to the last window, the "Libraries" window Figure 2.5

**Figure 2.4. CTT Export Notice and Installation Complete Message Window**
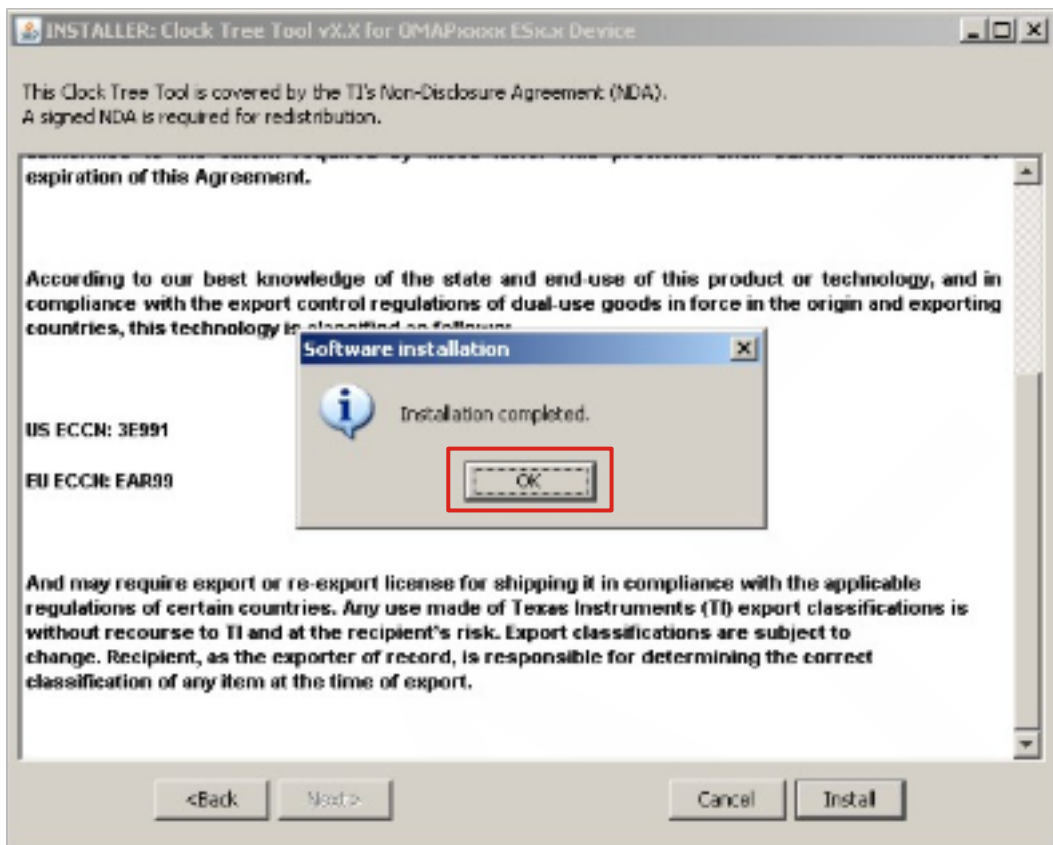


## 2.1.6. CTT Installation Step 6

The "Libraries" window Figure 2.5 identifies the libraries needed to run the Clock Tree Tool application. Click on "Exit" button to terminate the clock tree tool installer.

ℹ️ **Note**

Copy/Paste the URLs in your browser to go directrly to the needed downloads.

**Figure 2.5. CTT Installation Libraries Window**



The Clock Tree Tool is now installed! Please proceed to installing CTT JGraph Library.
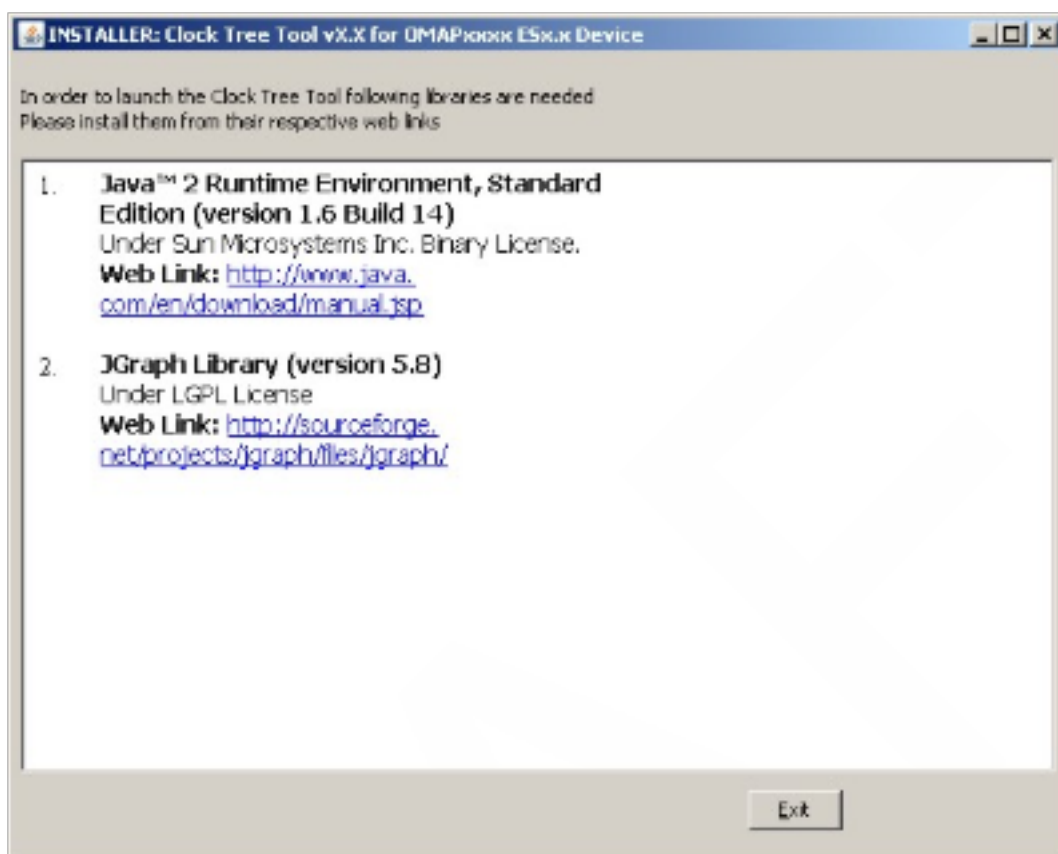
## 2.2. CTT JGraph Library (V5.8) Installation for use with Clock Tree Tool

### Note

Java™ Runtime Environment, Standard Edition (v 1.6 Build 17) **must** be installed.

Clock Tree Tool **must** be installed.

The JGraph library installation procedure is composed of following steps.

## 2.2.1. CTT JGraph Installation Step 1

On the "SOURCEFORGE.NET®" pageFigure 2.6 select the "jgraph-5.8.0.0-lgpl.jar" file for download, by clicking on the link. This leads to the "File download" window. To save the file to the local drive, click on the "Save" button.

### Caution

SOURCEFORGE®.NET Page is not compatable with Internet Explorer 6.0 or lower. If using IE 6 or lower, the webpage may appear differently. Please refer to another web source for downloading jgraph or click CTRL + F (for find) and search with key word "5.8". Do this several times, untill the webape scrolls itself to the needed version.

### Caution

CTT is not fully compatable with higher JGraph versions than 5.8. If higher version is installed, abnormal behaviour could appear.

**Figure 2.6. CTT JGraph Installation SOURCEFORGE®.NET Page**
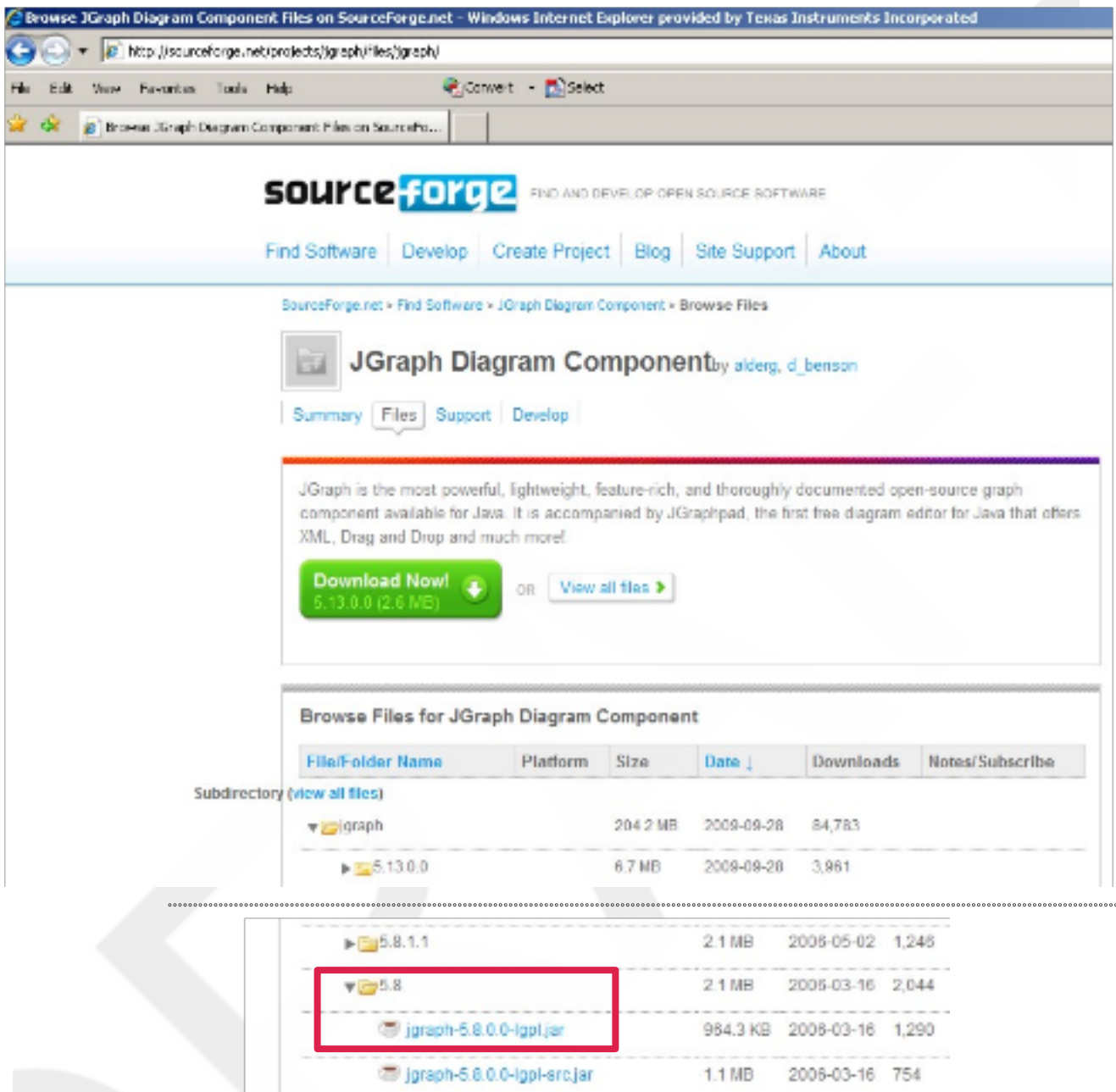


## 2.2.2. CTT JGraph Installation Step 2

Identify the destination directory for copying the downloaded file and click the "Select" button to start the file download. Once the download is complete, open the downloaded file by clicking the "Open" button in the "Download Complete" window (java -jar <jgraph.jar> for Linux users).

When opened, the JGraph installer will display a licence agreement window. Select the agreement button and click "Install". Proceed to Step 3.

## 2.2.3. CTT JGraph Installation Step 3

Select the "JGraphLib" directory within the Clock Tree Tool project directoryFigure 2.7 as the destination directory. The JGraph library will be installed in this directory.

**Figure 2.7. CTT JGraph Installation Directory Selection Window**
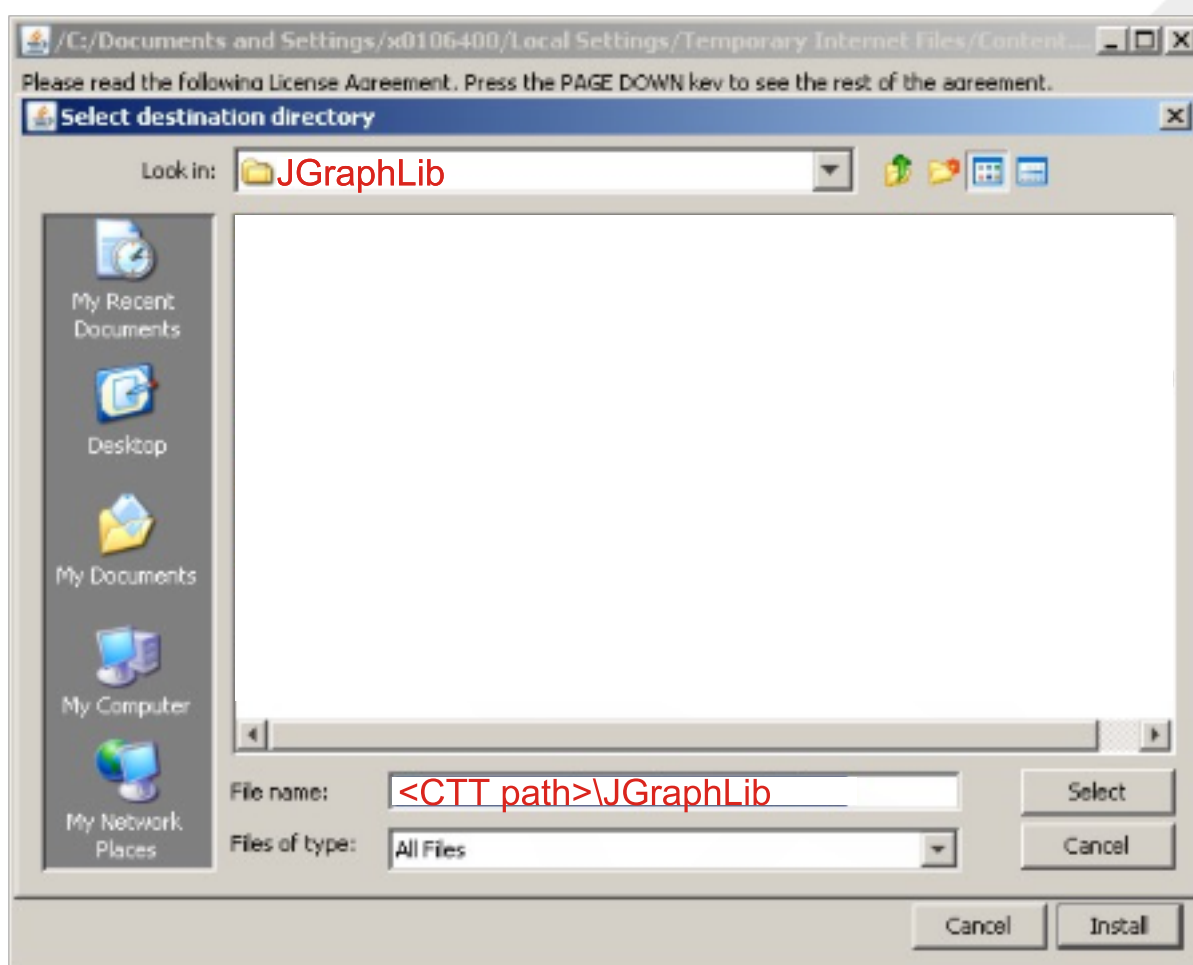


## 2.2.4. CTT JGraph Installation Step 4

Once the JGraph Library installation is complete and the Software Installation message window appears, click the "OK" button.

**Figure 2.8. CTT JGraph Installation Message Window**



## 2.3. CTT running when multiple versions of JRE are installed (Under MSWindows)

**⊗ Warning**

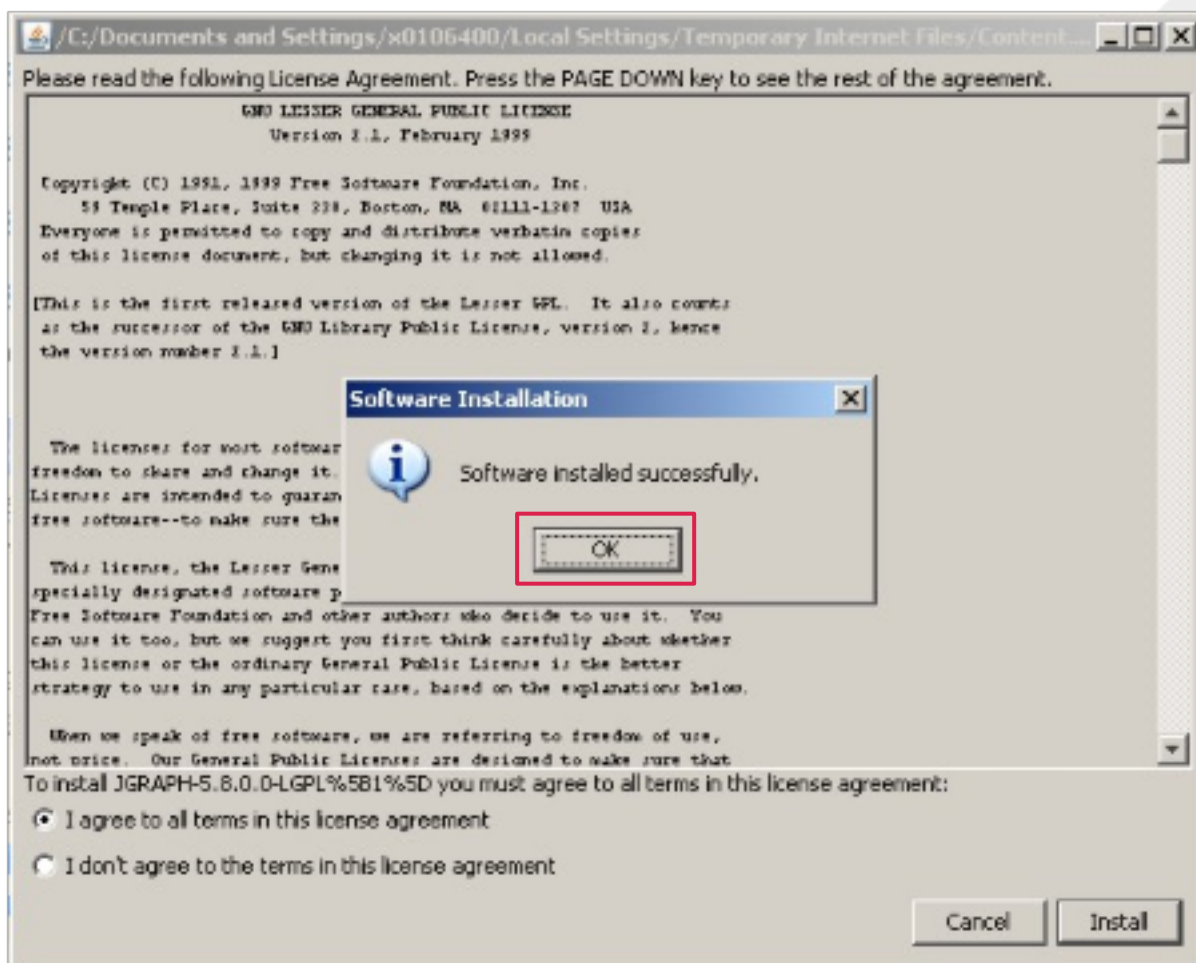JRE **version 1.6 Build 17** or higher is required. The default Java™ updater should install version 1.6 Build 17 or higher by default. If not, please refer to the Java™ update web page for downloading the 1.6 Build 17 or higher version. Then verify the installation by checking the version from the command line (type *java -version*).

**⊗ Warning**

CTT is capable of running with older JRE versions. Although this functionality, CTT performance and behaviour is unexpected if older JRE version is used.

When multiple versions of JRE are installed on the same machine, the default version for the java execution is the latest version.

For example, if the installed versions are:

- JRE version 1.4
- JRE version 1.5
- JRE version 1.6 (Default)

The current default version of JRE can be found by typing the following command on the command line:

```
java -version
```

The Clock Tree Tool requires JRE version 1.6 Build 17 for best performance. If you try to run the Clock tree tool (by double clicking on the CTTOMAP[full name].jar ) it should function properly if the required JRE version is installed.

CTT can also run with older JRE versions although it is not recommended. In order to run the Clock Tree Tool with the JRE version 1.5 following command line can be used from the directory containing the CTT jar file:

```
[full path to the JRE version 1.5 bin directory]\java -jar CTT-OMAP[full name].jar
```

For example:

```
C:\"Program Files"\Java\jre1.5.0_15\bin\java -jar CTT-OMAP4430ES1.1-v1.7.0.0b.jar
```

## Note

The "Quotes" in the path name are used when there are spaces in the path name.

Rather than typing this command every time to execute the Clock Tree Tool, a better option is to build a simple batch file. See below the example of a simple "CTT.bat" batch file:

```
@echo off
[full path to the JRE version 1.5 bin directory]\java -jar CTT-OMAP[full name].jar
```

Add this batch file to the directory containing the CTT jar file. Double clicking on the batch file will execute the CTT with correct JRE version 1.5.

## Warning

Running CTT with older JRE versions is not recommended.

# Chapter 3. Clock Tree Tool

## 3.1. CTT Overview

The Clock Tree Tool (CTT) is a Java™ based stand-alone application. It is an interactive clock tree configuration software for the device. It allows the user to:

- visualize the device clock tree
- interact with clock tree elements and view the effect on PRCM registers
- interact with the PRCM registers and view the effect on the device clock tree
- view a trace of all the device registers affected by the user interaction with clock tree

The advantage of the tool is that the user can visualize the device clock tree state on power-on reset and then customize the configuration of the clock tree for the specific use-case and identify the device register settings associated to that configuration.

Being an interactive visual tool, the CTT gives the user a global view of the device clock tree architecture and allows determining the exact register settings to obtain the specific configuration.

## 3.2. CTT System Requirements

- Requires Java™ JRE 1.6.0 or higher (Can be downloaded from http://java.sun.com).
- Has been tested for Microsoft WindowsXP™.
- The ideal screen resolution is 1280x1024.

## 3.3. CTT Running Requirements

1. Unzip the project zipped folder. (NOTE: Do not change the name of the unzipped folder, e.g., CTT-OMAP*XXXX* )

2. Double-click on the CTT-[version].jar file.

## 3.4. CTT Start-up and View Refresh Delays

## Note

The performance data given below is for the test machine used with AMD ™ Athlon™ 64 X2 Dual Core processor at 3.8GHz+ with 2 GB of RAM.

The start-up sequence of the CTT consists of reading an entire clock tree description database files. This would normally take about 30 seconds.

Similarly, the View Refresh function that updates the main view, covers the entire clock tree of the device and takes as well about 30 seconds.

## 3.5. CTT GUI (Graphical User Interface) Description

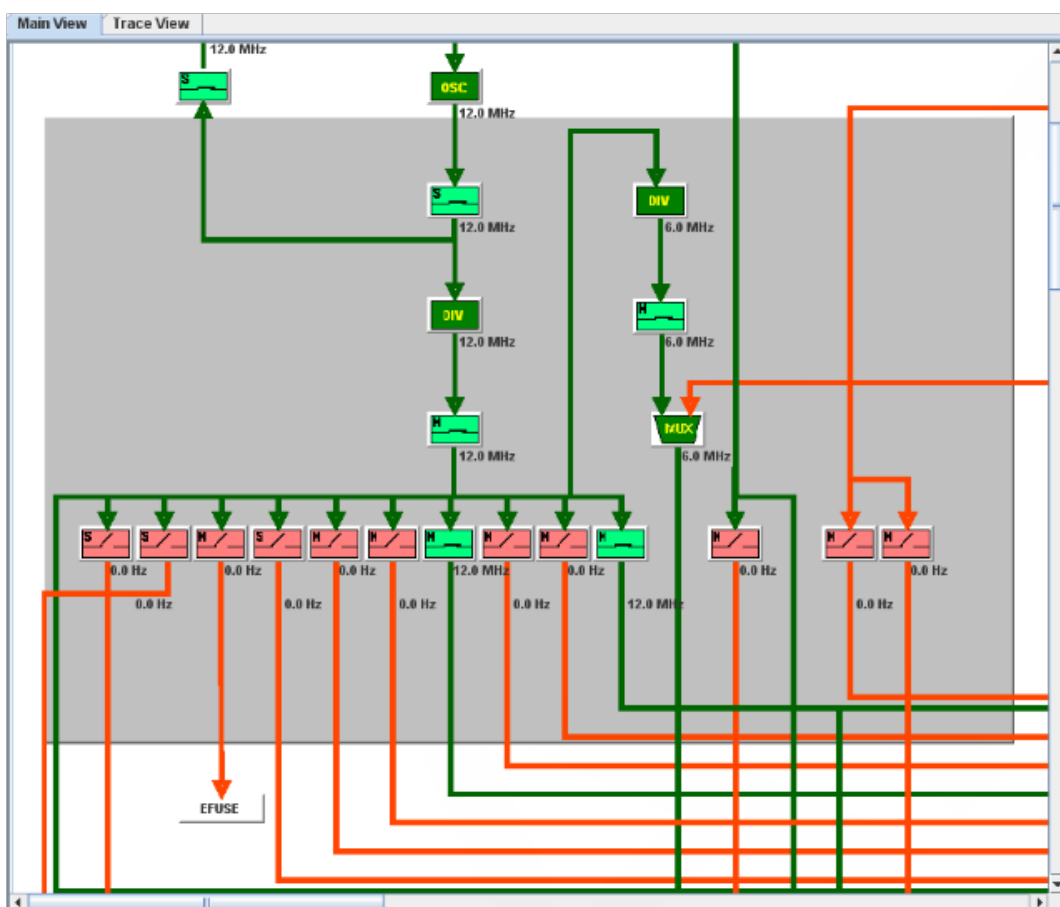## 3.5.1. CTT Views Description

**Figure 3.1. CTT Views**
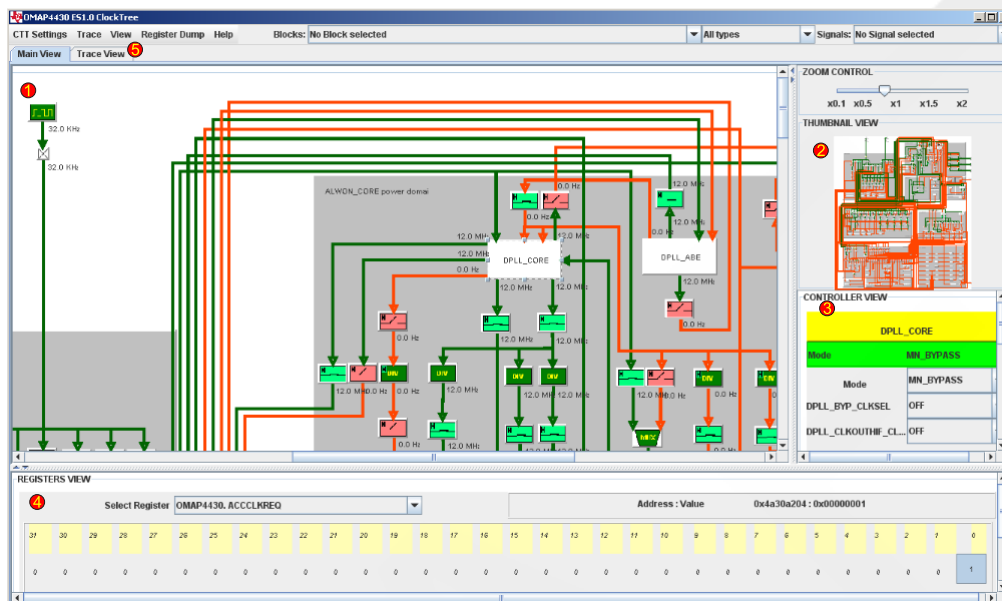


The CTT GUI is composed of 5 sub-views:

- Main View
- Thumbnail View
- Controller View
- Registers View
- Trace View

### 3.5.1.1. CTT Main View

The Main View presents a focused view of a section of the device clock tree.

**Figure 3.2. CTT Main View**



The device clock tree is represented as a tree structure composed of "nodes" or "blocks" (i.e., the rectangular elements) and the "links" or "signals" (i.e., the arrows). The direction of the signal identifies the source and the destination blocks of the signal. A block may be a source block to multiple blocks and may in turn have multiple source blocks connected to it.

The clock tree has following types of blocks:

- Crystal
- Clock Source
- Oscillator
- Clock Switch (Hardware/ Manual /Automatic)
- Multiplexer
- Divider
- DPLL
- Module

(Refer to the device TRM for the description of these blocks)

The user can use the slide bars on the right side and the bottom of the view to move up/down and left/right in the Main View, respectively. The view highlights the state of the blocks and the signals visually. For example, the state of a clock switch (Open/Close) is presented by a red open switch or a green close switch symbol. Similarly, the state of a clock signal (Active/ Gated) is highlighted by the signal being green or red.

## 3.5.1.2. CTT Thumbnail View

The Thumbnail View highlights a global view of the device clock tree. It also highlights the region of the clock tree visible in the Main View by a bounding rectangle. As the slide bars of the Main View are displaced the bounding rectangle in the Thumbnail View also moves accordingly.

**Figure 3.3. CTT Thumbnail View**
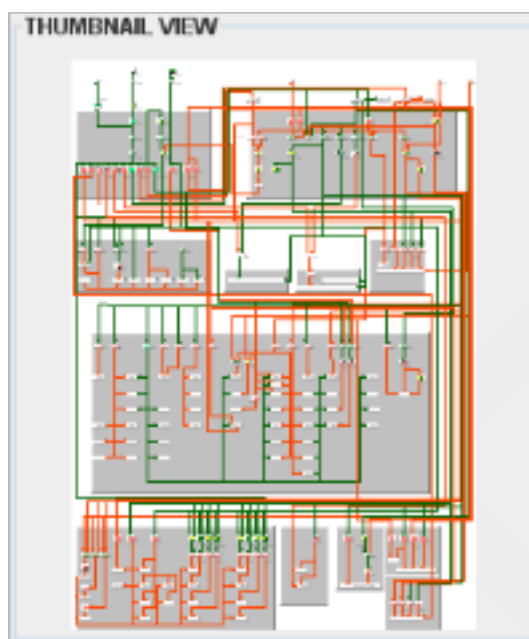


## 3.5.1.3. CTT Controller View

The Controller View highlights a signal or a block of the clock tree. The user selects (i.e., clicks on) the signal/ block in the Main View and it is highlighted in the Controller View. If a signal is selected, its current frequency is presented, whereas, if a block is selected, depending on the block type its parameters are presented.

**Figure 3.4. CTT Controller View**
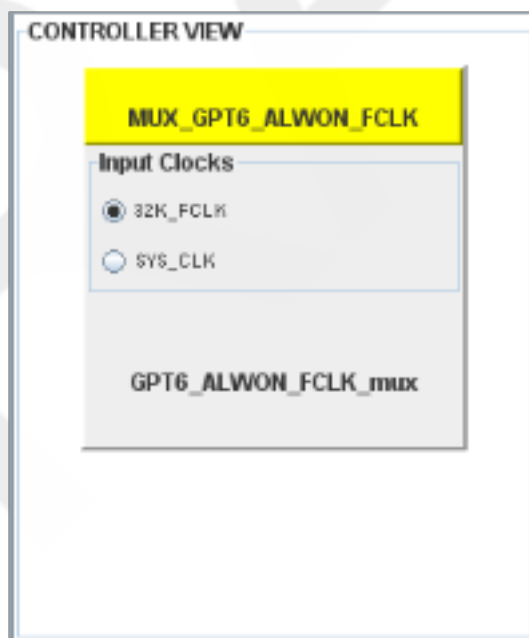


## 3.5.1.4. CTT Registers View

The Registers View is composed of a Register Selector list box, on the left hand side. The name of the currently selected register is highlighted in this box.

On the upper right hand side of the Register View is the Register Address/Value indicator. It presents the address and the current hexadecimal value of the register.

Below these two is a Register Bits view. The register bits view lists all the bits of the selected register (e.g., 0 to 31 bits for a 32 bits register of PRCM). Each bit is identified by the bit number (0 for the LSB). Below the bit number is the current value of the bit (1/0).

A toggle button below the bit number of the user configurable (i.e. read/write) bits allows the user to toggle the bit value. Pressing the button sets the bit value to 1 and in the released state the bit value is 0.

There is no button associated to the RESERVED bits of the register (i.e., the user cannot modify the states of these bits).

When the user selects a register in the Register Selector list box, its contents (i.e., bits and value) are highlighted in the Register Bits view and the Register Value indicator.

When the user changes a parameter of a block in the Controller View, the associated bitfield is updated in the register and the Register View highlights the affected register.

When the value of a bit/ bitfield of a register changes in the register view, the Trace view captures this change also.

# Note

When the user changes a parameter of a block which affects bitfields of more than one register, the Registers View only shows the last register updated. The Trace view shows the complete list of registers affected by this change.

## Figure 3.5. CTT Register View
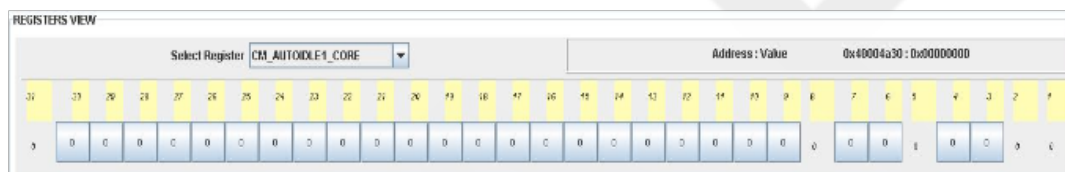


When user positions the pointer on the number of a register bit a pop-up displays the name of that bit.

## Figure 3.6. CTT Register View Pop-up
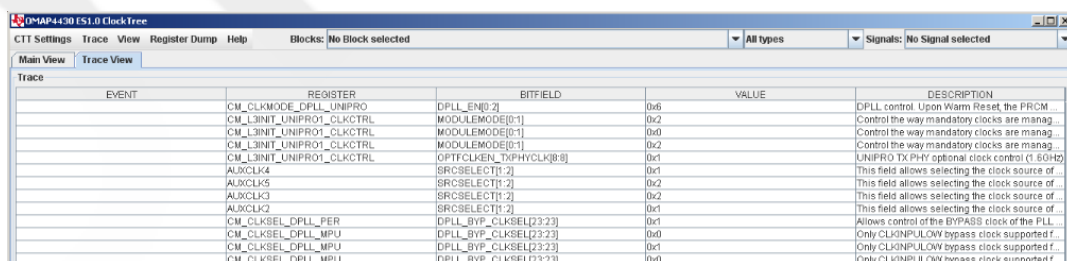


## 3.5.1.5. CTT Trace View

The Trace View is composed of a multi-column table. The successive changes of the register bitfields as a result of the user interaction with the clock tree (via the Controller View or the Register View) are recorded in the rows of the Trace View table.

This view allows the user to find a trace of all the register bitfields affected and the values associated to these bitfields, as a result of the current interactions.

## Figure 3.7. CTT Trace View



## 3.5.2. CTT Zoom Control

The Zoom Control allows the user to change the zoom level of the Main View. By default the zoom level is set to x1. The user can zoom in by shifting the slider to the right hand side (towards x2) and zoom-out by shifting the slider to the left hand side (towards x0.1)

**Figure 3.8. CTT Zoom Control**



### 3.5.3. CTT Search Bars

The CTT search bars allows the user to navigate directly to the desired block or signal within the main view. There are 3 bars. The first two are for searching and navigating to a particular block. The third one is for searching signals.

After selecting the desired block or signal, the main view will automatically scroll and highlight the selected (from the bars) block or signal.

**Figure 3.9. CTT Search Bars**
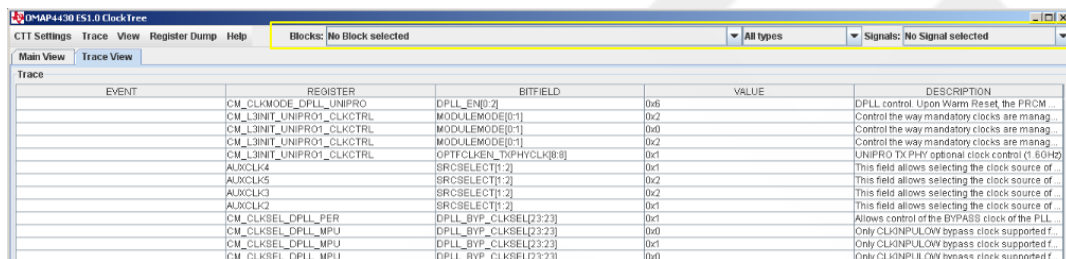


### 3.5.4. CTT Menu Commands Description

The CTT menu has following commands:

**Figure 3.10. CTT Menu**



1. CTT Settings
    1. Power-on Reset

2. Trace
    1. Reset

3. View
    1. Hide Others
    2. Display All
    3. Hide Frequency
    4. Display Frequency
    5. Refresh View
    6. Print View

4. Register Dump
    1. Load-in
    2. Dump-out

5. APN
    1. CTT Application Report

6. Help
    1. About Clock Tree Tool
    2. User Manual
    3. Licence Agreement
    4. Export Control Notice

### 3.5.4.1. CTT Settings

**1. Power-on Reset**Triggers a power-on reset for all the registers of the PRCM. All the registers are set to their reset values. As a result, the state of the clock tree is updated and reflects the state after power-on reset. (Note: When the CTT starts, the power-on reset is automatically triggered. Hence, the initial clock tree state is that of the device after power-on reset).

### 3.5.4.2. CTT Trace

**1. Reset**Resets (clears) the Trace View table.

### 3.5.4.3. CTT View

**1. Hide Others**
When a clock signal is selected in the Main View and this command is selected from the menu, the CTT hides all the clocks not associated to the selected clock. A clock is considered associated to another clock if it is directly/indirectly a parent/child of the clock.

**2. Display All**
This command is used to redisplay the entire clock tree from a partial view (as a result of the Hide Others command).

**3. Hide Frequency**
This command hides the frequency value of the clock signals in the Main View.

**4. Display Frequency**
This command displays the frequency values of the clock signals in the Main View.

**5. Refresh View**
This command refreshes the Main View representation of the clock tree. It is used if the clock tree representation is not correct and the view needs to be refreshed.

**6. Print View**
For a particular reason a user may want to print the tree onto an image. This image could be helpful if one needs to have it on paper, or just look at it without the need to load the CTT. This may also help when a user want to create a CTT configuration and print it. Then create another one and print it. This way the 2 or more print stamps can be compared and analyzed. When selected, the print option generated an image and saves it in the CTT install directory.

### 3.5.4.4. CTT Register Dump

The register dump menu allows the user to either configure the registers of the PRCM (used in the CTT) to specific settings given in a file or to write the current values of the registers of the CTT to a file.

## Note

The Register Dump may be used to read-in the registers dump file generated by the Register Dump script (GEL in Code Composer Studio or CMM in Lauterbach) or by any function respecting the format described below. This allows to read the current state of clock tree at any break-point in the code and obtain a visual representation of its state in the CTT. A gel and a cmm files can be found in *<CTT-Install-path>/Scripts/*

**1. Load-in** This menu allow to read-in the contents of a file, in order to set the values of the register of the CTT. The tool can read two different file formats as given below.
**1.1 Load Format:** ADD DATA. This menu selection allows to read the file in which each line consists of two hex values in following format : ADDRESS_OF_REGISTER DATA_VALUE_OF_REGISTER
**1.2 Load Format:** [ADD] DATA. This menu selection allows to read the file in which each line consists of two hex values in following format :[ADDRESS_OF_REGISTER] DATA_VALUE_OF_REGISTER

**2. Dump-out** This menu allow to read-in the contents of a file, in order to set the values of the register of the CTT. The tool can read two different file formats as given below.
**2.1 Dump Format:** ADD DATA. This menu selection allows to write to file in format where each line consists of two hex values as given below : ADDRESS_OF_REGISTER DATA_VALUE_OF_REGISTER
**2.2 Dump Format:** [ADD] DATA. This menu selection allows to write to file in format where each line consists of two hex values as given below : [ADDRESS_OF_REGISTER] DATA_VALUE_OF_REGISTER

## Note

The ADD DATA format is saved to / read from file with .rd1 extension.

The [ADD] DATA format is saved to / read from file with .rd2 extension.

### 3.5.4.5. CTT APN

**1. CTT Application Report**This option opens the CTT Application Report document. (For Linux users, navigate to the "Docs" folder and open document with appropriate application manually.)

### 3.5.4.6. CTT Help

**1. About Clock Tree Tool**This section highlights the different elements of the clock tree and the associated controller description.

**2. User Manual**This option opens the user manual document. (For Linux users, navigate to the "Docs" folder and open document with appropriate application manually.)

**3. Licence Agreement**This option displays the licence agreement window.

**4. Export Control Notice**This option displays the export control notice window.

## 3.6. CTT Blocks

This section highlights the different types of blocks that model the clock tree behavior, in the CTT.

> **Note**
>
> Any modification of the Block parameters in the Controller View affects the associated register bitfields. The Register View switches to the most recently updated register, while all the bitfield value changes are also added to the Trace View.
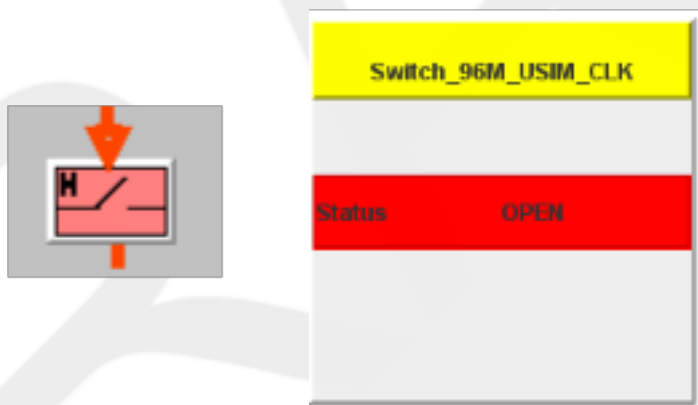
### 3.6.1. CTT Clock Switch Block

A clock switch allows the clock gating control (i.e., enable/ disable) within the branches of the clock tree. Essentially, three different types of switches may be defined:

1. Hardware Switch

2. Manual Switch

3. Auto Switch

### 3.6.1.1. CTT Hardware Switch

**Figure 3.11. CTT Hardware Switch**



The hardware switch is controlled by hardware gating conditions:

- The derived clock is inactive.
- All modules receiving the derived clock, are inactive.
- All switches receiving the derived clock, are gated (open).

The user has no control over this switch. It is automatically closed when the hardware gating conditions are satisfied.
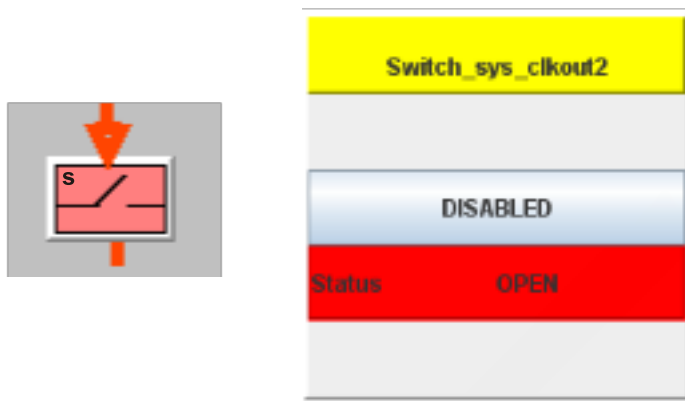
> **Note**
>
> A derived clock is the clock at the output of the switch.

## 3.6.1.2. CTT Manual Switch

**Figure 3.12. CTT Manual Switch**

The manual switch is software controlled by setting or clearing the enable bits in corresponding registers (Generally applicable to module functional clocks). The user can enable or disable the switch using the button in the controller view.
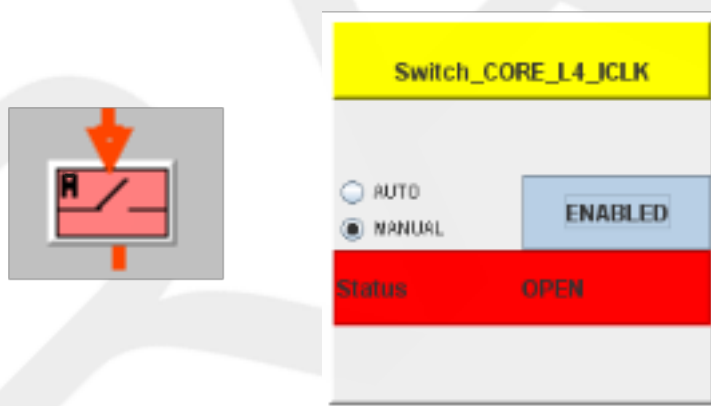
The derived clock from the switch may be connected to multiple modules and can have one or more ENABLE bits associated, to request this clock.

The switch gating condition is:

- All the associated clock ENABLE bits for this clock are cleared to 0.

## 3.6.1.3. CTT Auto Switch

**Figure 3.13. CTT Auto Switch**

The auto switch is a software/ hardware controlled switch. The user can either manually (through software control) enable/disable the derived clock or set the switch to auto mode.

In the auto mode the clock is controlled by hardware gating conditions. Hence, when ever the gating conditions are satisfied the clock is automatically disabled and when any of the gating conditions is not satisfied the clock is automatically enabled by the hardware. In this case no software control of clocks is necessary.

The manual (software) control clock gating condition is:

- All the associated clock ENABLE bits for this clock are cleared to 0.

The hardware control clock gating conditions are:

- All the associated clock ENABLE bits and clock AUTO bits for this clock are set to 1.
- The derived clock is not requested by any module (i.e., the module is inactive).

> ## Note
>
> Both the clock gating conditions of the auto mode must be satisfied for the derived clock to be gated automatically

The user can use the switch in manual mode by clicking on the MANUAL check box.

In this case when the push-button on the right side is in ENABLED state, all the associated clock ENABLE bits are set and the switch is closed. Similarly if the push-button is in DISABLED state, all the clock ENABLE bits are cleared to 0 and the switch is open.

The user can set the switch to auto mode using following sequence:

1. Push the push-button to ENABLED state, to set all clock ENABLE bits to 1.

2. Click on the AUTO check box to set all the clock AUTO bits to 1.

In the auto mode, the switch will automatically close when any of its gating conditions is not satisfied.

## 3.6.2. CTT Divider Block

A clock divider allows the clock frequency division. The output clock frequency is the frequency of the input clock divide by the division factor set in the divider.
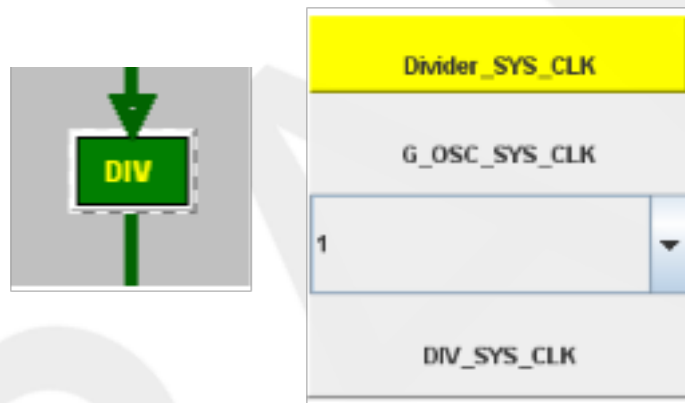
The user can select the division factor by clicking on the drop-down list.

## Note

If the divider has a fixed divide factor (i.e., the software can not change the divide factor) then the drop-down list contains only one division factor.

**Figure 3.14. CTT Divider Block**



## 3.6.3. CTT MUX Block

A clock mux allows selecting from multiple source clocks for the derived clock. The user can select the source clock by clicking on the check box corresponding to one of the multiple source clocks in the Controller view.
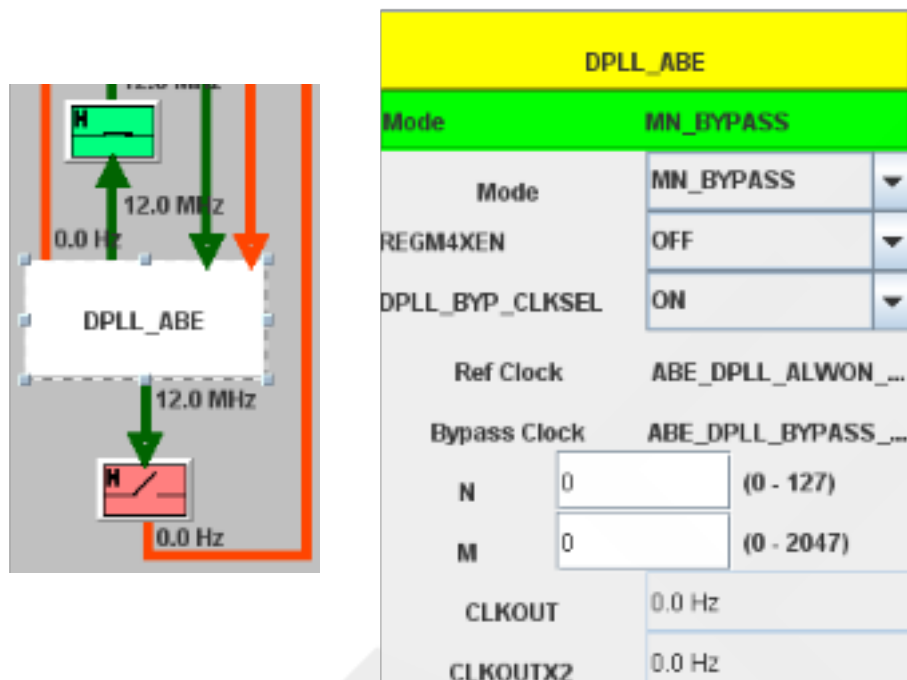
The currently selected source clock is identified in the Controller view.

**Figure 3.15. CTT MUX Block**
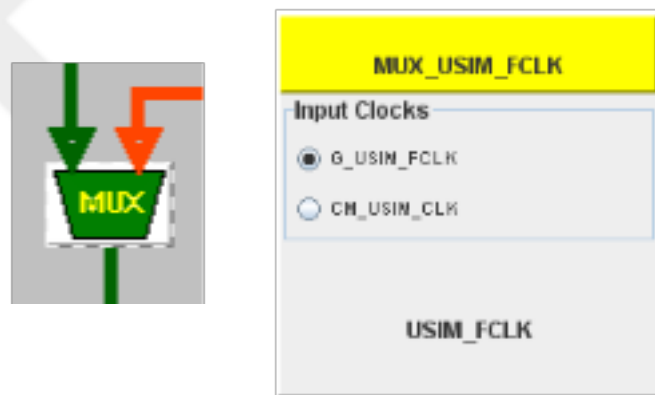


## 3.6.4. CTT DPLL Block

A DPLL block receives source clocks and in turn generates the clocks for the device. Refer to the TRM for details about the DPLL.

## Figure 3.16. CTT DPLL Block



The user must follow the following sequence to configure the DPLL:

1. If the DPLL is in LOCKED state, set it to one of the UNLOCKED states (e.g., LOW POWER STOP state), by selecting the mode in the "Mode" drop-down list.

2. Set the M and N parameters by typing the values in the corresponding edit boxes. NOTE: After editing the value of the parameter in the edit box, ENTER key must be pressed so that the new value is accepted by the tool.

3. Select the output divide factor M2, etc., by clicking on the associated drop-down list.

4. Switch the DPLL to the LOCKED mode by clicking on the "Mode" drop-down list and selecting the mode.

Once the DPLL is in LOCKED state the CLKOUT, CLKOUTX2 and the output clock frequencies (displayed after the output divide factors) will be updated.

DPLLs can also have options from the controller to select bypass clocks, 4xen mode, CLOCKOUTIF, and sd-div modes. For more information about these functionality please refer to the PRCM TRM.

## 3.6.5. CTT Module Block

A module block represents the destination modules, such as I2C, MCSI, McBSP etc. A module receives functional and interface clocks. It may be active or inactive. It can also be in enabled, auto, or disabled module mode. Module can also have optional functional clocks associated to it.

If a module has only Active / Idle functionality, the user can switch a module to ACTIVE or IDLE state and only the MODULE STATE drop down menu will be displayed inside the controller.

If a module has MODULE MODE and MODULE STATE functionality, the user must select the mode of the module, then the module state.

If a module has Optional Functional clock functionality, the user may enable opt clocks as well.

> ## Note
>
> In the device, there are various combinations of module functionality. A given module can have one or more at the same time.

Moreover, In basics, the clocks associated to module function as follows:

1. Optional functional clock is running whenever the OptFclken bit is set to 1, and it is not concerned by the module state (idle/active).

2. Module mode associated clocks are automatically gated if ModuleMode is set to "Disabled" and this is the module reaches idle state.

3. When ModuleMode is set to "Enabled" functional clock is automatically un-gated. The interface clock is automatically gated/un-gated based on the module idle/active transition.

4. Module mode associated clocks are automatically gated/un-gated when ModuleMode is set to "Auto" based on the idle/active transition of the module. "Auto" option is available only for modules with interface idle protocol associated clock(s).

For more information about module mode, module state, and optional clocks associated to modules, please refer to PRCM TRM.

**Figure 3.17. CTT Module Block**
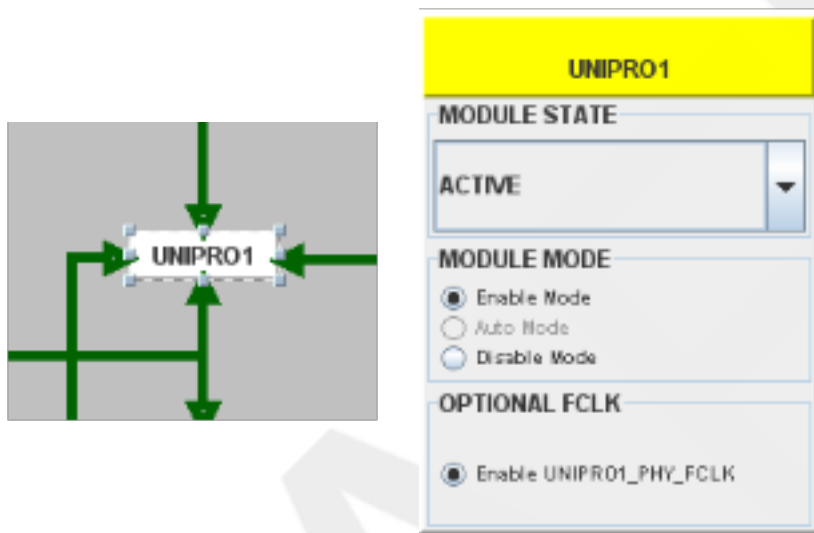


## 3.7. CTT Limitations And Bugs

This section identifies the known limitations and bugs of the clock tree tool.

### 3.7.1. CTT Thumbnail View Tracking Rectangle Error

The blue tracking rectangle on the thumbnail view, identifies the visible region of the clock tree in the main view. It may show following behavior:

- The tracking rectangle is invisible.
- Two tracking rectangles are visible
- Tracking rectangle is not in synchronization with the main view.

**Solution:**Under development.

### 3.7.2. CTT Refresh Limitation

Currently, in a given case a refresh limitation may occur and an associated block controller may not display properly the selected functionality. For this reason, a user can unclick/click on the not displayed properly functionality twice to eliminate the refreshing limitation.

### 3.7.3. CTT Status Bits

Currently, the clock tree tool does not present the status bits associated to the clocks and module states

### 3.7.4. CTT Register Controller

Currently, there may be cases where certain block register controllers (when clicked) will not update properly the associated controller. For this reason, a user can use the controller instead which will update the register.

### 3.7.5. CTT Power Domains Behavior

> ## Note
>
> CTT presents the clock behavior with the assumption that the associated power domains are in ON power state.

Clock Tree Tool does not model the behavior of the clocks when the power domains are in RETENTION or OFF power states. It also does not model the SLEEP and WAKE-UP related clock behavior.

Currently, the clock tree tool does not present the status bits associated to the clocks and module states