

Intel[®] ME Firmware Integrated Clock Controller (ICC) Tool

Tools User Guide

January 2012

Revision 1.0

Intel Confidential



Notice: This document contains information on products in the design phase of development. The information here is subject to change without notice. Do not finalize a design with this information.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: <http://www.intel.com/design/literature.htm>

I2C is a two-wire communications bus/protocol developed by Philips. SMBus is a subset of the I2C bus/protocol and was developed by Intel. Implementations of the I2C bus/protocol may require licenses from various entities, including Philips Electronics N.V. and North American Philips Corporation.

Intel® Active Management Technology (Intel® AMT) requires activation and a system with a corporate network connection, an Intel® AMT-enabled chipset, network hardware and software. For notebooks, Intel AMT may be unavailable or limited over a host OS-based VPN, when connecting wirelessly, on battery power, sleeping, hibernating or powered off. Results dependent upon hardware, setup and configuration. For more information, visit <http://www.intel.com/technology/platform-technology/intel-amt>.

WARNING: Altering clock frequency and/or voltage may: (i) reduce system stability and useful life of the system and processor; (ii) cause the processor and other system components to fail; (iii) cause reductions in system performance; (iv) cause additional heat or other damage; and (v) affect system data integrity. Intel has not tested, and does not warranty, the operation of the processor beyond its specifications. Intel assumes no responsibility that the processor, including if used with altered clock frequencies and/or voltages, will be fit for any particular purpose. For more information, visit [Overclocking Intel Processors](#)

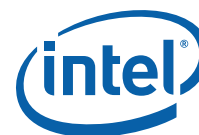
Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families: Go to: http://www.intel.com/products/processor_number

Code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2013, Intel Corporation. All rights reserved.



Contents

1	Introduction	5
1.1	Terminology.....	5
1.2	Reference Documents	5
2	ICC Tools	7
2.1	Command Line Interface.....	7
2.2	Examples.....	9
2.2.1	Example 1 - Get Clock Capabilities	9
2.2.2	Example 2 - Get Intel Clock Range Definition Record	9
2.2.3	Example 3 - Get OEM Clock Range Definition Record	11
2.2.4	Example 4 - Get Lock	13
2.2.5	Example 5 - Get Profiles	13
2.2.6	Example 6 - Get Record (Intel)	13
2.2.7	Example 7 - Get Record (Current)	14
2.2.8	Example 8 - Read Register (All)	15
2.2.9	Example 9 - Read Register (Names)	19
2.3	Error and Status Messages.....	20
2.3.1	Clock Commander Tool Error and Status Messages	20
2.3.2	Boot Status.....	21

Tables

1-1	Terminology.....	5
1-2	Reference Documents	5
2-1	CCT Error and Status Messages	20
2-2	ICC Boot Status Errors	21
2-3	ICC Boot Status Informational Messages.....	22



Revision History

Revision Number	Description	Revision Date
0.5	<ul style="list-style-type: none">Initial release	March 2011
0.6	<ul style="list-style-type: none">Added flags for WUOB command	May 2011
0.7	<ul style="list-style-type: none">Add new command gcemParameters changed for commands sl, sce, wuob and smrCommand capability changed for command smrRemove temporary UOB in command grUpdate some examples in section 2.3Remove boot status errors "TempUobViolatedCikRangeLimits" and "TempUobApplyingFailure" in table 2.2Add one new CCT status message in section 2.4	June 2011
0.8	<ul style="list-style-type: none">Change Intel® 7 Series Express Chipset to Intel® 7 Series/C216 Chipset Family	September 2011
0.9	<ul style="list-style-type: none">Update command usage and its NOTESRemove the feature of SMBus and its commands, smr and smwRemove the command rrd and gcemRemove selector, perm, from command grRemove the flag for ICC Record, postuobThe file cct.ini is no longer used while removing Control via SMBUSExample of output by commands are updatedTo add two CCT error and status messages in Table 2-1To remove DOS_WAIT, BAD_NONCE, and DOS_WAIT_BAD_NONCE from Table 2-1To remove DENIED_AUTO_LOCKED from Table 2-1	February 2012
0.91	<ul style="list-style-type: none">Rename Intel® ME Firmware Integrated Clock Controller (ICC)	April 2012
0.92	<ul style="list-style-type: none">Remove SMBus related. No SMBus supported on ME9.0	June 2012
0.93	<ul style="list-style-type: none">Minor changes to note on ICC Tool output example (Section 2.2) will be different on Intel® ME8.0, Intel® ME8.1, Intel® ME9.0 and Intel® ME9.5	August 2012
1.00	<ul style="list-style-type: none">Finalize to Rev1.0	January 2013

S



1 Introduction

The purpose of the document is to provide guidance on the usage of the tools provided for Intel® ME Firmware Integrated Clock Controller (ICC) included within the Intel® Management Engine (Intel® ME) firmware kit.

1.1 Terminology

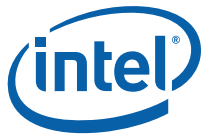
Table 1-1. Terminology

Acronym or Term	Definition
API	Application Programming Interface
BIOS	Basic Input Output System
CCT	Clock Commander Tool
CCTwin	Windows* command line version of the Clock Commander Tool
CPU	Central Processing Unit
DLL	Dynamic Link Library
FITC	Flash Image Tool
FW	Firmware
HECI (deprecated)	Host Embedded Controller Interface
ICCS	Integrated Clock Controller Services
Intel® ME	Intel Management Engine
Intel® MEI	Intel Management Engine Interface (formerly HECI)
PCH	Platform Controller Hub
Permanent UOB	UOB that is applied on every boot.
UOB	Update on Boot. An record of ICC registers setting that are applied on the next platform boot.

1.2 Reference Documents

Table 1-2. Reference Documents

Document	Document No. / Location
Lynx Point SPI Programming Guide	FW release kit
Lynx Point Intel® Management Engine Firmware Bring Up Guide	FW release kit
Lynx Point Platform Controller Hub (PCH) External Design Specification (EDS)	Please contact your FAE for availability.





2 ICC Tools

This document covers the usage of the Clock Commander Tool (CCT) included in the `..\Tools\ICC_tools\` directory. Details on other tools can be found in the tools user guides included in the other tools directories contained within the firmware kit.

The CCT tools included in the Lynx Point firmware release kit are designed for Lynx Point based platforms only. These tools will not function on other legacy platforms.

2.1 Command Line Interface

CCT.exe and CCTwin.exe support the following command line options. To view all of the supported options, run the application with no arguments or with the ? option. The command syntax for the CCT tool is CCT [options] command [arguments].

The Windows* version of the tool - CCTwin.exe - requires that the Intel® MEI driver is loaded for it to function.

The available options are:

/v0 - verbose level 0. This is the default mode and provides the smallest amount of information.

/v1 - verbose level 1. This is the debug mode and includes additional debug information including the raw Intel® MEI message information.

The available CCT command are:

CCT	gcc [no arguments]
	gcdr [selector] [OEM index]
	gl [no arguments]
	sl [registers to lock] [params]
	sce [clock enables] [clock enables mask] [params]
	gp [no arguments]
	sp [profile number]
	gr [selector]
	rr [buffered ¹ register or registers name to read]
	wr [register name] = [register value]
	wuob [flags] [register name] = [register value]

NOTES:

1. Option for registers that have two stages (buffered & active). The "buffered" option reads the value yet to propagate to the active stage. When reading a single stage register where there is no buffered value, the active value is returned.

gcc:

Gets ICC clock capabilities



gcdr:

Gets combined clock range definition record used by FW. Could be run with "oem" selector. If so then requested clock range definition record for current profile is returned. If a record index is specified (e.g. oem 1) then requested clock range definition record for profile #1 is returned.

gl:

Show which registers are locked and cannot be written after EOP.

sl:

Locks specified registers. The registers to be locked can be specified as symbolic names or as 32 bit register masks. A single register can be specified or a list of registers can be specified. This command would typically be used by BIOS developers. This command is to specify which registers will be unlocked/locked after EOP. This command will not work after the BIOS sends the End of Post Intel® MEI message. A flag option "noresp" can be used if CCT doesn't want a response from Intel® ME FW.

sce:

Enables or disables selected PCI clock outputs. The clock enables argument is a 32 bit value which specifies the clock output settings. The clock enables mask argument is a 32 bit value which specifies which clock outputs will be enabled or disabled. This command would typically be used by BIOS developers. This command will not work after the BIOS sends the End of Post Intel® MEI message. A flag option "noresp" can be used if CCT doesn't want a response from Intel® ME FW.

gp:

Gets the currently used ICC profile number.

sp:

Sets the ICC profile to the number specified in the profile number argument. This command will not work after the BIOS sends the End of Post Intel® MEI message.

gr:

Gets the ICC record specified in the selector argument. The available selectors are:

intel - Intel record

oem - OEM record

preuob - platform boot time record pre UOB

current - current record

rr:

Reads registers based on the register argument. The register can be specified as a list of decimal or hexadecimal offsets or a list of symbolic names. When specified as a list offsets and symbolic names can be mixed. Registers can also be specified as a range in which case only numbers can be used. This command also accepts the buffered option for registers that have two stages.

wr:

This command writes ICC registers based on the register offset and value arguments. The arguments need to be specified in the form of a pair in the form of *register offset = register value*. the register offset can be specified as a number or as a symbolic name.

wuob:

Write a UOB record. The flag for this command is invalid - invalidation request for the UOB record. If no flags are used, a permanent UOB is created or invalidated.



2.2 Examples

<The output shown below will be a little different between Intel® ME8.0, Intel® ME8.1, Intel® ME9.0 and Intel® ME9.5>

2.2.1 Example 1 - Get Clock Capabilities

```
C:\cct>cctwin.exe gcc
```

```
Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.
```

```
icc_hw_version_number = 0004.0000
icc_hw_sku = ENHANCED
icc_boot_status_report [0x00c00000]:
    boot event: "SetClockEnablesReceived"
    boot event: "LockReceived"
```

```
HECI CMD Status = 0x00000000 (SUCCESS)
```

2.2.2 Example 2 - Get Intel Clock Range Definition Record

```
C:\cct>cctwin.exe gcdr intel
```

```
Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.
```

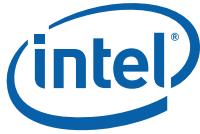
```
clock_id = 1[MODIV 1]

clock_usage = {} -> NOT USED
frequency_min           = 135.0000 MHz
frequency_max           = 135.0000 MHz
ssc_change_allowed      = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 0
ssc_spread_percent_max  = 0.50 %

clock_id = 2[MODIV 2]

clock_usage = {} -> NOT USED
frequency_min           = 98.8558 MHz
frequency_max           = 100.0000 MHz
ssc_change_allowed      = 1
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 1
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 1
ssc_spread_percent_max  = 0.50 %

clock_id = 3[MODIV 3]
```



```
clock_usage = {} -> NOT USED
frequency_min      = 99.4819 MHz
frequency_max      = 100.0000 MHz
ssc_change_allowed = 1
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 1
ssc_spread_percent_max = 0.50 %
```

```
clock_id = 4[MODIV 4]
```

```
clock_usage = {} -> NOT USED
frequency_min      = 135.0000 MHz
frequency_max      = 135.0000 MHz
ssc_change_allowed = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 0
ssc_spread_percent_max = 0.00 %
```

```
clock_id = 5[MODIV 5]
```

```
clock_usage = {} -> NOT USED
frequency_min      = 135.0000 MHz
frequency_max      = 135.0000 MHz
ssc_change_allowed = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 0
ssc_spread_percent_max = 0.00 %
```

```
clock_id = 6[MODIV 6]
```

```
clock_usage = {} -> NOT USED
frequency_min      = 96.0000 MHz
frequency_max      = 96.0000 MHz
ssc_change_allowed = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 0
ssc_spread_percent_max = 0.00 %
```

```
clock_id = 7[MODIV 7]
```

```
clock_usage = {} -> NOT USED
frequency_min      = 89.1641 MHz
frequency_max      = 89.1641 MHz
ssc_change_allowed = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 0
```



```

ssc_spread_percent_max           = 0.00 %

clock_id = 8[MODIV 8]

clock_usage = {} -> NOT USED
frequency_min           = 135.0000 MHz
frequency_max           = 135.0000 MHz
ssc_change_allowed      = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 0
ssc_spread_percent_max  = 0.50 %

```

HECI CMD Status = 0x00000000 (SUCCESS)

2.2.3 Example 3 - Get OEM Clock Range Definition Record

C:\cct>cctwin.exe gcdr OEM 0

Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.

```

clock_id = 1[MODIV 1]

clock_usage = {} -> NOT USED
frequency_min           = 135.0000 MHz
frequency_max           = 135.0000 MHz
ssc_change_allowed      = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 0
ssc_spread_percent_max  = 0.50 %

clock_id = 2[MODIV 2]

clock_usage = {} -> NOT USED
frequency_min           = 100.0000 MHz
frequency_max           = 100.0000 MHz
ssc_change_allowed      = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 0
ssc_spread_mode_control_halt_allowed = 0
ssc_spread_percent_max  = 0.00 %

clock_id = 3[MODIV 3]

clock_usage = {BCLK, DMI, PEG, PCIe, PCI33, SATA, USB3}
frequency_min           = 100.0000 MHz
frequency_max           = 100.0000 MHz
ssc_change_allowed      = 1

```



```
ssc_spread_mode_control_up_allowed      = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed    = 1
ssc_spread_mode_control_halt_allowed    = 1
ssc_spread_percent_max                  = 0.50 %

clock_id = 4[MODIV 4]

clock_usage = {} -> NOT USED
frequency_min      = 135.0000 MHz
frequency_max      = 135.0000 MHz
ssc_change_allowed = 0
ssc_spread_mode_control_up_allowed      = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed    = 1
ssc_spread_mode_control_halt_allowed    = 0
ssc_spread_percent_max                  = 0.00 %

clock_id = 5[MODIV 5]

clock_usage = {} -> NOT USED
frequency_min      = 135.0000 MHz
frequency_max      = 135.0000 MHz
ssc_change_allowed = 0
ssc_spread_mode_control_up_allowed      = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed    = 1
ssc_spread_mode_control_halt_allowed    = 0
ssc_spread_percent_max                  = 0.00 %

clock_id = 6[MODIV 6]

clock_usage = {} -> NOT USED
frequency_min      = 96.0000 MHz
frequency_max      = 96.0000 MHz
ssc_change_allowed = 0
ssc_spread_mode_control_up_allowed      = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed    = 1
ssc_spread_mode_control_halt_allowed    = 0
ssc_spread_percent_max                  = 0.00 %

clock_id = 7[MODIV 7]

clock_usage = {} -> NOT USED
frequency_min      = 89.1641 MHz
frequency_max      = 89.1641 MHz
ssc_change_allowed = 0
ssc_spread_mode_control_up_allowed      = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed    = 1
ssc_spread_mode_control_halt_allowed    = 0
ssc_spread_percent_max                  = 0.00 %

clock_id = 8[MODIV 8]
```



```

clock_usage = {} -> NOT USED
frequency_min          = 135.0000 MHz
frequency_max          = 135.0000 MHz
ssc_change_allowed     = 0
ssc_spread_mode_control_up_allowed = 0
ssc_spread_mode_control_center_allowed = 0
ssc_spread_mode_control_down_allowed = 1
ssc_spread_mode_control_halt_allowed = 0
ssc_spread_percent_max = 0.00 %

```

HECI CMD Status = 0x00000000 (SUCCESS)

2.2.4 Example 4 - Get Lock

```
C:\cct>cctwin.exe gl
```

```

Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.

```

REGISTERS TO BE UNLOCKED AFTER EOP:

```

SSCDIVINTPHASE_DMI100
SSCTRIPARAM_DMI100
SSCCTL_DMI100
SSCDIVINTPHASE_PCHPCIE100
SSCTRIPARAM_PCHPCIE100
SSCCTL_PCHPCIE100
MDYNCTL

```

HECI CMD Status = 0x00000000 (SUCCESS)

2.2.5 Example 5 - Get Profiles

```
C:\cct>cctwin.exe gp
```

```

Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.

```

```

number_of_icc_profiles      = 2
oem_boot_profile_number    = 0
icc_profile_is_selected_by  = oem (strap)
current_boot_profile_index  = 0

```

HECI CMD Status = 0x00000000 (SUCCESS)

2.2.6 Example 6 - Get Record (Intel)

```
C:\cct>cctwin.exe gr intel
```

```

Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.

```

```

FLAGS [0x00012080]
record_length      = 128

```



```
REGISTERS
SECURITY0          = 0xffffffffcd
SECURITY1          = 0xffffffff9ff
SECURITY2          = 0xfffffffffff
BIAS0              = 0x2ab05530
SBEPCTL            = 0x00020110
SSCCTL_DCLK135     = 0x00000000
SSCANACTL_DCLK135  = 0xff8f61ff
SSCANACTL_DMI100   = 0xff8f61ff
SSCANACTL_PCHPCIE100 = 0xff8f61ff
SSCCTL_DCLKBEND    = 0x00000008
SSCANACTL_DCLKBEND = 0xff8f61ff
SSCANACTL_VGACLK   = 0xff8f61ff
SSCANACTL_USB96    = 0xff8f61ff
SSCANACTL_REF14    = 0xff8f61ff
SSCANACTL_27S      = 0xff8f61ff
DBUFF0_CLKOUT_DP_SSC = 0x00000f01
```

HECI CMD Status = 0x00000000 (SUCCESS)

2.2.7 Example 7 - Get Record (Current)

```
C:\cct>cctwin.exe gr current
```

Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.

```
FLAGS [0x0001206c]
record_length      = 108
```

```
REGISTERS
SSCDIVINTPHASE_DMI100      = 0x00000000
SSCTRIPARAM_DMI100        = 0x00000000
SSCCTL_DMI100              = 0x00000001
SSCDIVINTPHASE_PCHPCIE100 = 0x00000032
SSCTRIPARAM_PCHPCIE100    = 0x12404038
SSCCTL_PCHPCIE100         = 0x00000000
DIV_PCI33                  = 0x00030203
DIV_FLEX4824               = 0x00030103
OCKEN                      = 0x7d9c0f8f
MDYNCTL                    = 0x00000004
SEFLXNP                    = 0x00009999
SEPCICLKBP                 = 0x00099999
DCOSS                      = 0x00000400
SECOSS                     = 0x00002516
MCSS                       = 0x00000001
PLLRCS                     = 0x00011114
ICCCTL                     = 0x00000018
PMPCI                      = 0x00000000
PM1SRCCLK                  = 0x76543210
PM2SRCCLK                  = 0x00000098
```

HECI CMD Status = 0x00000000 (SUCCESS)



2.2.8 Example 8 - Read Register (All)

```
C:\cct>cctwin.exe rr all
```

```
Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.
```

```
SECURITY0          = 0xffffffffcd
SECURITY1          = 0xffffffff9ff
SECURITY2          = 0xfffffffffff
BIAS0              = 0x2ab05530
BIAS1              = 0x000000f2
BIAS2              = 0x00000000
BIASMISC           = 0x00000088
CLKPATH            = 0x070f7f99
MODDIV_FB          = 0x00000134
LCPLL0             = 0x00000000
LCPLL1             = 0x00000000
LCPLL2             = 0x00005560
LCPLL3             = 0x00000000
LCPLL4             = 0x00000000
LCPLLMON           = 0x00000000
OSC0               = 0x0000005c
SFR0               = 0x00020301
MONPORT0           = 0xe0000000
MONPORT1           = 0x00000000
MUXTOP            = 0x00000000
VISACTL0           = 0x00000000
VISACTL1           = 0x00000000
VISACTL2           = 0x00000000
CBMISC             = 0x00000000
SBEPCTL            = 0x00020110
MONPORT2           = 0x00000000
CMNRSTFSM          = 0x00001d4c
SSCDIVINTPHASE_DCLK135 = 0x00000024
SSCDITHPHASE_DCLK135 = 0x00000000
SSCTRIPARAM_DCLK135 = 0x27708028
SSCCTL_DCLK135     = 0x00000000
SSCAUXDIV_DCLK135  = 0x00000021
SSCMISC_DCLK135    = 0x00000001
SSCTRIPARAMEXTRA_DCLK135 = 0x000029c5
SSCRSVD_DCLK135    = 0x00000000
SSCANACTL_DCLK135  = 0xff8f61ff
SSCRSTFSM_DCLK135  = 0x004d0012
SSCDFXPHSPLN0_DCLK135 = 0x00000000
SSCDFXPHSPLN1_DCLK135 = 0x00000000
SSCDFXMISC0_DCLK135 = 0x02000000
SSCDFXMISC1_DCLK135 = 0x00000000
SSCDFXMON_DCLK135  = 0x00000000
SSCDFXMISR0_DCLK135 = 0x00400001
SSCDFXMISR1_DCLK135 = 0x00000000
SSCDFXCNT_DCLK135  = 0x00000000
SSCVISA0_DCLK135   = 0x00000000
SSCVISA1_DCLK135   = 0x00000000
SSCVISA2_DCLK135   = 0x00000000
SSCDIVINTPHASE_DMI100 = 0x00000000
```



SSCDITHPHASE_DMI100	= 0x00000000
SSCTRIPARAM_DMI100	= 0x00000000
SSCCTL_DMI100	= 0x00000001
SSCAUXDIV_DMI100	= 0x00000029
SSCMISC_DMI100	= 0x00000001
SSCTRIPARAMEXTRA_DMI100	= 0x000029c5
SSCRSVD_DMI100	= 0x00000000
SSCANACTL_DMI100	= 0xff8f61ff
SSCRSTFSM_DMI100	= 0x004d0012
SSCDFXPHSPLN0_DMI100	= 0x00000000
SSCDFXPHSPLN1_DMI100	= 0x00000000
SSCDFXMISC0_DMI100	= 0x02000000
SSCDFXMISC1_DMI100	= 0x00000000
SSCDFXMON_DMI100	= 0x00000000
SSCDFXMISR0_DMI100	= 0x00400001
SSCDFXMISR1_DMI100	= 0x00000000
SSCDFXCNT_DMI100	= 0x00000000
SSCVISA0_DMI100	= 0x00000000
SSCVISA1_DMI100	= 0x00000000
SSCVISA2_DMI100	= 0x00000000
SSCDIVINTPHASE_PCHPCIE100	= 0x00000032
SSCDITHPHASE_PCHPCIE100	= 0x00000000
SSCTRIPARAM_PCHPCIE100	= 0x12404038
SSCCTL_PCHPCIE100	= 0x00000000
SSCAUXDIV_PCHPCIE100	= 0x00000029
SSCMISC_PCHPCIE100	= 0x00000001
SSCTRIPARAMEXTRA_PCHPCIE100	= 0x000029c5
SSCRSVD_PCHPCIE100	= 0x00000000
SSCANACTL_PCHPCIE100	= 0xff8f61ff
SSCRSTFSM_PCHPCIE100	= 0x004d0012
SSCDFXPHSPLN0_PCHPCIE100	= 0x00000000
SSCDFXPHSPLN1_PCHPCIE100	= 0x00000000
SSCDFXMISC0_PCHPCIE100	= 0x02000000
SSCDFXMISC1_PCHPCIE100	= 0x00000000
SSCDFXMON_PCHPCIE100	= 0x00000000
SSCDFXMISR0_PCHPCIE100	= 0x00400001
SSCDFXMISR1_PCHPCIE100	= 0x00000000
SSCDFXCNT_PCHPCIE100	= 0x00000000
SSCVISA0_PCHPCIE100	= 0x00000000
SSCVISA1_PCHPCIE100	= 0x00000000
SSCVISA2_PCHPCIE100	= 0x00000000
SSCDIVINTPHASE_DCLKBEND	= 0x00000024
SSCDITHPHASE_DCLKBEND	= 0x00000000
SSCTRIPARAM_DCLKBEND	= 0x27708028
SSCCTL_DCLKBEND	= 0x00000008
SSCAUXDIV_DCLKBEND	= 0x00000021
SSCMISC_DCLKBEND	= 0x00000001
SSCTRIPARAMEXTRA_DCLKBEND	= 0x000029c5
SSCRSVD_DCLKBEND	= 0x00000000
SSCANACTL_DCLKBEND	= 0xff8f61ff
SSCRSTFSM_DCLKBEND	= 0x004d0012
SSCDFXPHSPLN0_DCLKBEND	= 0x00000000
SSCDFXPHSPLN1_DCLKBEND	= 0x00000000
SSCDFXMISC0_DCLKBEND	= 0x02000000
SSCDFXMISC1_DCLKBEND	= 0x00000000
SSCDFXMON_DCLKBEND	= 0x00000000



SSCDFXMISR0_DCLKBEND	= 0x00400001
SSCDFXMISR1_DCLKBEND	= 0x00000000
SSCDFXCNT_DCLKBEND	= 0x00000000
SSCVISA0_DCLKBEND	= 0x00000000
SSCVISA1_DCLKBEND	= 0x00000000
SSCVISA2_DCLKBEND	= 0x00000000
SSCDIVINTPHASE_VGACLK	= 0x000015ba
SSCDITHPHASE_VGACLK	= 0x00000000
SSCTRIPARAM_VGACLK	= 0x27708028
SSCCTL_VGACLK	= 0x00000008
SSCAUXDIV_VGACLK	= 0x00000021
SSCMISC_VGACLK	= 0x00000001
SSCTRIPARAMEXTRA_VGACLK	= 0x000029c5
SSCRSVD_VGACLK	= 0x00000000
SSCANACTL_VGACLK	= 0xff8f61ff
SSCRSTFSM_VGACLK	= 0x004d0012
SSCDFXPHSPLN0_VGACLK	= 0x00000000
SSCDFXPHSPLN1_VGACLK	= 0x00000000
SSCDFXMISC0_VGACLK	= 0x02000000
SSCDFXMISC1_VGACLK	= 0x00000000
SSCDFXMON_VGACLK	= 0x00000000
SSCDFXMISR0_VGACLK	= 0x00400001
SSCDFXMISR1_VGACLK	= 0x00000000
SSCDFXCNT_VGACLK	= 0x00000000
SSCVISA0_VGACLK	= 0x00000000
SSCVISA1_VGACLK	= 0x00000000
SSCVISA2_VGACLK	= 0x00000000
SSCDIVINTPHASE_USB96	= 0x00000834
SSCDITHPHASE_USB96	= 0x00000000
SSCTRIPARAM_USB96	= 0x12700000
SSCCTL_USB96	= 0x00000008
SSCAUXDIV_USB96	= 0x00000021
SSCMISC_USB96	= 0x00000001
SSCTRIPARAMEXTRA_USB96	= 0x000029c5
SSCRSVD_USB96	= 0x00000000
SSCANACTL_USB96	= 0xff8f61ff
SSCRSTFSM_USB96	= 0x004d0012
SSCDFXPHSPLN0_USB96	= 0x00000000
SSCDFXPHSPLN1_USB96	= 0x00000000
SSCDFXMISC0_USB96	= 0x02000000
SSCDFXMISC1_USB96	= 0x00000000
SSCDFXMON_USB96	= 0x00000000
SSCDFXMISR0_USB96	= 0x00400001
SSCDFXMISR1_USB96	= 0x00000000
SSCDFXCNT_USB96	= 0x00000000
SSCVISA0_USB96	= 0x00000000
SSCVISA1_USB96	= 0x00000000
SSCVISA2_USB96	= 0x00000000
SSCDIVINTPHASE_REF14	= 0x000012b8
SSCDITHPHASE_REF14	= 0x24489123
SSCTRIPARAM_REF14	= 0x12700000
SSCCTL_REF14	= 0x00000008
SSCAUXDIV_REF14	= 0x00000031
SSCMISC_REF14	= 0x00000001
SSCTRIPARAMEXTRA_REF14	= 0x000029c5
SSCRSVD_REF14	= 0x00000000



SSCANACTL_REF14	= 0xff8f61ff
SSCRSTFSM_REF14	= 0x004d0012
SSCDFXPHSPLN0_REF14	= 0x00000000
SSCDFXPHSPLN1_REF14	= 0x00000000
SSCDFXMISC0_REF14	= 0x02000000
SSCDFXMISC1_REF14	= 0x00000000
SSCDFXMON_REF14	= 0x00000000
SSCDFXMISR0_REF14	= 0x00400001
SSCDFXMISR1_REF14	= 0x00000000
SSCDFXCNT_REF14	= 0x00000000
SSCVISA0_REF14	= 0x00000000
SSCVISA1_REF14	= 0x00000000
SSCVISA2_REF14	= 0x00000000
SSCDIVINTPHASE_27S	= 0x00000000
SSCDITHPHASE_27S	= 0x00000000
SSCTRIPARAM_27S	= 0x00000000
SSCCTL_27S	= 0x00000009
SSCAUXDIV_27S	= 0x00000029
SSCMISC_27S	= 0x00000001
SSCTRIPARAMEXTRA_27S	= 0x000029c5
SSCRSVD_27S	= 0x00000000
SSCANACTL_27S	= 0xff8f61ff
SSCRSTFSM_27S	= 0x004d0012
SSCDFXPHSPLN0_27S	= 0x00000000
SSCDFXPHSPLN1_27S	= 0x00000000
SSCDFXMISC0_27S	= 0x02000000
SSCDFXMISC1_27S	= 0x00000000
SSCDFXMON_27S	= 0x00000000
SSCDFXMISR0_27S	= 0x00400001
SSCDFXMISR1_27S	= 0x00000000
SSCDFXCNT_27S	= 0x00000000
SSCVISA0_27S	= 0x00000000
SSCVISA1_27S	= 0x00000000
SSCVISA2_27S	= 0x00000000
DIV_27NS	= 0x00428417
DIV_27S	= 0x000a8103
DIV_PCI33	= 0x00030203
DIV_FLEX4824	= 0x00030103
DIV_MECLK	= 0x00220105
DIV_VECLK	= 0x00228203
DBUFF0_CLKOUT_ITPXD	= 0x00000f01
DBUFF1_CLKOUT_ITPXD	= 0x7f07070f
DBUFF0_CLKOUT_DMI	= 0x00000f01
DBUFF1_CLKOUT_DMI	= 0x7f07070f
DBUFF0_CLKOUT_SRC0	= 0x00000f01
DBUFF1_CLKOUT_SRC0	= 0x7f07070f
DBUFF0_CLKOUT_SRC1	= 0x00000f01
DBUFF1_CLKOUT_SRC1	= 0x7f07070f
DBUFF0_CLKOUT_SRC2	= 0x00000f01
DBUFF1_CLKOUT_SRC2	= 0x7f07070f
DBUFF0_CLKOUT_SRC3	= 0x00000f01
DBUFF1_CLKOUT_SRC3	= 0x7f07070f
DBUFF0_CLKOUT_SRC4	= 0x00000f01
DBUFF1_CLKOUT_SRC4	= 0x7f07070f
DBUFF0_CLKOUT_SRC5	= 0x00000f01
DBUFF1_CLKOUT_SRC5	= 0x7f07070f



```

DBUFF0_CLKOUT_SRC6           = 0x00000f01
DBUFF1_CLKOUT_SRC6           = 0x7f07070f
DBUFF0_CLKOUT_SRC7           = 0x00000f01
DBUFF1_CLKOUT_SRC7           = 0x7f07070f
DBUFF0_CLKOUT_PEGA           = 0x00000f01
DBUFF1_CLKOUT_PEGA           = 0x7f07070f
DBUFF0_CLKOUT_PEGB           = 0x00000f01
DBUFF1_CLKOUT_PEGB           = 0x7f07070f
DBUFF0_CLKOUT_DP_SSC         = 0x00000f01
DBUFF1_CLKOUT_DP_SSC         = 0x7f07070f
DBUFF0_CLKOUT_DP_NSSC        = 0x00000f01
DBUFF1_CLKOUT_DP_NSSC        = 0x7f07070f
ICCMTR                        = 0x00000000
OCKEN                         = 0x7d9c0f8f
MDYNCTL                       = 0x00000004
RSVD_0C                       = 0x00000000
RSVD_10                       = 0x00000000
RSVD_14                       = 0x00000000
RSVD_18                       = 0x00000000
RSVD_1C                       = 0x00000000
SEOBEN                        = 0x00000f8f
SEFLXNP                       = 0x00009999
SEPCICLKBP                    = 0x00099999
RSVD_2C                       = 0x00000000
RSVD_30                       = 0x00000000
RSVD_34                       = 0x00000000
RSVD_38                       = 0x00000000
RSVD_3C                       = 0x00000000
DCOSS                         = 0x00000400
SECOSS                        = 0x00002516
MCSS                          = 0x00000001
PLLRCS                        = 0x00011114
RSVD_50                       = 0x00000000
RSVD_54                       = 0x00000000
RSVD_58                       = 0x00000000
RSVD_5C                       = 0x00000000
ICCCTL                        = 0x00000018
ICC_SPARE                     = 0x00000000
PMPCI                         = 0x00000000
PM1SRCCLK                     = 0x76543210
PM2SRCCLK                     = 0x00000098
ICCSFBV                       = 0x0000000c
ICCSBV                        = 0x00000002
RSVD_7C                       = 0x00000000

```

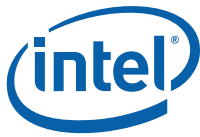
HECI CMD Status = 0x00000000 (SUCCESS)

2.2.9 Example 9 - Read Register (Names)

```
C:\cct>cctwin.exe rr DIV_PCI33 OCKEN
```

Intel (R) Clock Commander Tool Version: 9.0.0.7046
Copyright (C) 2012 Intel Corporation. All rights reserved.

```
DIV_PCI33                     = 0x00030203
```



OCKEN = 0x7d9c0f8f

HECI_CMD_Status = 0x00000000 (SUCCESS)

2.3 Error and Status Messages

2.3.1 Clock Commander Tool Error and Status Messages

When a command is executed the Clock Commander Tool will display status and error messages to indicate the result of the operations. The messages and their definitions are listed in the following table.

Table 2-1. CCT Error and Status Messages

CCT Message	Definition
SUCCESS	The command executed successfully.
FAILURE	The command failed to execute.
INVALID_OPTION	An invalid option was specified for the command.
INVALID_COMMAND	The command entered was invalid.
INVALID_ARGUMENT	The argument entered was invalid.
REGISTER_OFFSET_OUT_OF_RANGE	The register offset entered was outside the allowable range.
TOO_FEW_ARGUMENTS	Arguments missing from the command.
HECI_INITIALIZATION_FAILED	Initialization of the Intel® MEI interface failed.
HECI_READ_FAILED	A read from the Intel® MEI interface failed.
HECI_WRITE_FAILED	A write to the Intel® MEI interface failed.
INVALID_RESPONSE	The command received an invalid response.
INVALID_FUNCTION	An invalid function was sent to the FW.
INVALID_PARAMS	A command failed due to invalid parameters.
FLASH_WEAR_OUT_VIOLATION	FW is indicating a flash wear out violation.
FLASH_CORRUPTION	FW is indicating that the flash is corrupted.
PROFILE_NOT_SELECTABLE_BY_BIOS	The ICC profile is not selectable by BIOS. It is selectable by a soft strap.
TOO_LARGE_PROFILE_INDEX	The profile sent by the command exceeds the number of profiles present in the flash.
NO_SUCH_PROFILE_IN_FLASH	The profile sent by the command does not exist in the flash.
CMD_NOT_SUPPORTED_AFTER_END_OF_POST	A command was attempted that is not allowed after end of post is received from the BIOS.
NO_SUCH_RECORD	A command attempted to access a non-existent record.
NO_SUCH_REGISTER	A command attempted to access a non-existent register.
NO_SUCH_TARGET_ID	A command attempted to access a non-existent target ID.
TOO_LARGE_REGISTER_INDEX	The register index is outside the allowable range.

**Table 2-1. CCT Error and Status Messages**

CCT Message	Definition
TOO_LARGE_UOB_RECORD	A write UOB command failed because the UOB exceeded the allowable size.
REGISTER_IS_LOCKED	Access to the ICC register is denied because it is locked.
FUNCTION_NOT_SUPPORTED_AFTER_EOP_OVE R_THIS_HECI	A command was attempted that is not allowed after end of post is received from the BIOS.
FUNCTION_NOT_SUPPORTED_OVER_SMBUS	A command is sent that is not supported over the SMBus.
UOB_RECORD_IS_ALREADY_INVALID	This error occurs when CCT attempts to invalidate a UOB that is already invalid.
ONE_UOB_RECORD_IS_ALREADY_VALID	An attempt is made to create a UOB when one is already valid.
OCKEN_MASK_VIOLATION	An attempt is made to write to the OCKEN register that violates the clock enables mask settings.
SUCCESS_OCKEN_AUTO_LOCKED	The OCKEN register was successfully auto locked by FW.
RANGE_VIOLATION_FREQ_TOO_HIGH_CLK[x]	A command failed because the frequency exceeded the allowable range.
RANGE_VIOLATION_FREQ_TOO_LOW_CLK[x]	A command failed because the frequency exceeded the allowable range.
SSC_MODE_CHANGE_NOT_SUPPORTED_CLK[x]	A command failed because a change to the spread spectrum mode is not supported for that clock.
AS EXPECTED, RESPONSE FROM Intel® ME FW NOT RECEIVED	No response from Intel® ME FW received

2.3.2 Boot Status

The Clock Commander Tool Command Get Clock Capabilities (gcc) returns an ICC boot status report which provides an indication of the status of integrated clock control after the system has booted. The possible results of the boot status are shown in the following table.

Table 2-2. ICC Boot Status Errors

Boot Status Message	Definition
IccBootRecoveryFailure	There was some failure during the ICC boot recovery.
RecoveredFromIccWdtTimeout	FW detected a watch dog timer expiration.
DisqualifiedIccProfile	The BIOS ICC profile was disqualified. This could be due to the FW not receiving the DRAM init done message.
IccProfileSelectionFailure	Selection of the ICC profile failed.
IccProfileIndexOutOfRange	The selected ICC profile exceeds the number of profiles contained in flash.
OemPltParamsBlockInvalid	The ICC NVAR in flash has an invalid format.
IccCrdrCreationFailure	Creation of the clock range definition record failed.
OemClkRangeMinViolation	The OEM record violates one of the Intel minimum ranges.



Table 2-2. ICC Boot Status Errors

Boot Status Message	Definition
OemClkRangeMaxViolation	The OEM record violates one of the Intel maximum ranges.
OemSprPrcntMaxViolation	The OEM record violates the Intel spread spectrum range for one of the clocks.
IntelRecordApplyingFailure	Application of the Intel record failed.
OemRecordViolatedClkRangeLimits	The OEM record violates the range limits for one of the clocks.
OemRecordApplyingFailure	Application of the OEM record failed.
PermUobViolatedClkRangeLimits	The permanent UOB is outside the clock ranges for one of the clocks.
PermUobApplyingFailure	Application of the permanent UOB failed.
SusramRecoveryFailure	FW was not able to successfully restore all the contents from SUSRAM to flash.
IntelCRDRSkuReducedEnhancedUpperRange	FW has detected that Intel® ME clk OC might occur on Enhanced SKU and thus upper range for Intel® ME clk must be changed to basic.
IntelCRDRSkuReducedExtremeRanges	FW has detected that Intel® ME clk OC might occur on Extreme SKU and thus both ranges for Intel® ME clk must be changed to basic.
OemRecordViolatedMEClkRestrictions	FW has detected that Intel® ME clk is trying to be routed to CLK4 in the OEM Record.
UobRecordViolatedMEClkRestrictions	FW has detected that Intel® ME clk is trying to be routed to CLK4 in the UOB Record.

Table 2-3. ICC Boot Status Informational Messages

Boot Status Message	Definition
GetIccProfileReceived	Get ICC profile command received.
SetClockEnablesReceived	Received set clock enables command from BIOS.
LockReceived	Received the lock ICC registers command from BIOS.
CmosBatteryRemoved	FW detected that the CMOS battery was removed.
InvalidatedUobRecord	The UOB record has been invalidated.

§