



Intel[®] Quark[™] SoC X1000 Software

Package Version: 1.0.1

Release Notes

22 May 2014



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>.

Any software source code reprinted in this document is furnished for informational purposes only and may only be used or copied and no license, express or implied, by estoppel or otherwise, to any of the reprinted source code is granted by this document.

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/%5Fnumber/>

Code Names are only for use by Intel to identify products, platforms, programs, services, etc. ("products") in development by Intel that have not been made commercially available to the public, i.e., announced, launched or shipped. They are never to be used as "commercial" names for products. Also, they are not intended to function as trademarks.

Intel, the Intel logo, and Quark are trademarks of Intel Corporation in the U.S. and/or other countries.

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

Copyright © 2014, Intel Corporation. All rights reserved.



Contents

1.0	Description of Release	4
1.1	New Features in Release 1.0.1	4
1.2	Features	4
1.2.1	BIOS/Firmware	4
1.2.2	Bootloader	5
1.2.3	Linux* Operating System (OS)	5
1.2.4	OpenOCD	6
1.3	Limitations	6
1.4	Unplanned Functionality	7
1.5	Component Versions	7
1.5.1	Packages	7
1.5.2	BIOS/Firmware Version	7
1.6	Related Documentation	7
1.7	Licensing	8
2.0	Known Issues	9
3.0	Resolved Issues	20

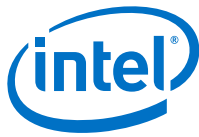
Tables

1	Related Documentation	7
2	License Files	8
3	Known Issue Summary	9
4	Resolved Issue Summary	20

Revision History

Date	Revision	Description
22 May 2014	002	General updates for software release 1.0.1 in the following sections: <ul style="list-style-type: none"> Section 1.1, “New Features in Release 1.0.1” on page 4. Section 1.3, “Limitations” on page 6. Section 1.7, “Licensing” on page 8. Section 2.0, “Known Issues” on page 9. Section 3.0, “Resolved Issues” on page 20.
04 March 2014	001	First public release of document.





1.0 Description of Release

This document describes extensions and deviations from the release functionality described in the documentation for the Intel® Quark™ SoC X1000 (formerly codenamed Clanton).

This release is called: Package Version: 1.0.1

Intel® Quark™ SoC X1000 Software supports the following Form Factor Reference Design boards (FFRDs):

- Customer Reference Boards:
 - Kips Bay (Fab C, green PCB)
 - Galileo (Fab D, blue PCB)
- Intel® Quark™ SoC X1000 Industrial/Energy Reference Design, “Cross Hill”
- Intel® Quark™ SoC X1000 Transportation Reference Design, “Clanton Hill”
- Intel-only System Validation Platform (SVP), “Clanton Peak”

For instructions on building and running the release software, see the Intel® Quark™ SoC X1000 Board Support Package (BSP) Build and Software User Guide (see [Table 1](#)).

These release notes also include known issues with third-party or reference platform components that affect the operation of the software.

1.1 New Features in Release 1.0.1

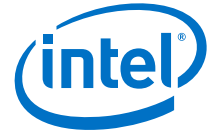
- UART driver enables DMA capability (transmit only) in Intel® Quark™ SoC X1000 which increases performance for CAN and RS-485 operation
- Toolchain update for 'lock prefix' bug in LMT CPU core
- Some bug fixes for eADC and other minor updates
- Cross Hill boards: SLCAN support, single channel
- Cross Hill boards: TPM support in BIOS
- Adds support for Galileo Gen2 board

1.2 Features

Features supported in this release are listed in the following subsections.

1.2.1 BIOS/Firmware

- Recovery:
 - Force recovery support (jumper/strap to force the system into recovery mode)
 - Secure recovery support (recovery capsules must be validly signed for Secure SKUs)



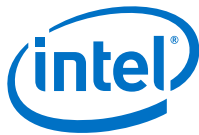
- Update:
 - Secure update support (update capsules must be validly signed for Secure SKUs)
- Secure Lock Down build support for secure SKUs. Includes -DSECURE_ID build option for creating image for secure SKUs. This restricts the boot options from EDKII (USB/SD/UEFI Shell boot are not allowed).
- Security features:
 - Protected BIOS Range registers, thus protecting more SPI flash regions.
 - SMI protection of SPI flash (secure SKUs only). Prevents non-EDKII code from updating SPI flash.
- ECC scrubbing (memory patrol scrubbing) disabled regardless of fuse setting
- Switch from SPI flash mapped platform data to ACPI objects for platform ID, MAC addresses, and serial number
- Secure boot using Root Of Trust ROM when using a secure SKU Intel® Quark™ SoC X1000
- Boot device selection:
 - SD boot
 - USB (OHCI/EHCI) boot
 - Payload boot (application in legacy SPI flash)
 - EFI Shell
- UEFI 2.3.1 compliant
- ACPI 5.0

1.2.2 Bootloader

- Secure boot Root of Trust when using a secure SKU Intel® Quark™ SoC X1000
- Isolated Memory Region (IMR) protection of compressed Linux* kernel before executing kernel
- Bootloader executed as payload from SPI flash
- Ability to load kernel and root-filesystem from SPI flash
- U-Boot memory tests ported and included

1.2.3 Linux* Operating System (OS)

- IsADC and eADC (including calibration) optional plug-in for timer-based sampling trigger
- Drivers for Transportation Reference Design (Clanton Hill)
 - STMicroelectronics* LIS331DLH Accelerometer Driver
 - Audio Subsystem Driver
 - Analog AD7298 ADC Driver
- Thermal Driver
- HE910 3G Driver
- WiFi Driver:
 - Intel® Centrino® Wireless-N 135 (also provides Bluetooth via USB)
 - Intel® Centrino® Advanced-N 6205 (Dual Band WiFi, 2.4 and 5 GHz)



- I²C* interface
- IMR protection of kernel, text, and data sections
- Kernel logic to parse platform data specific to Clanton Peak, Industrial/Energy Reference Design (Cross Hill), and Transportation Reference Design (Clanton Hill)
- Ethernet
 - Two Ethernet interfaces: Clanton Peak, Industrial/Energy Reference Design (Cross Hill), and Transportation Reference Design (Clanton Hill)
 - One Ethernet interface: Kips Bay and Intel® Galileo board
- GPIOs fully programmable as input or output from kernel gpiolib
- SPI master interface x 2
- USB OHCI/EHCI port x 2
- USB device
- SD master interface
- ECC updates configurable at runtime through /sysfs interface
- Small embedded user-space busybox based system < 2 megabytes compressed

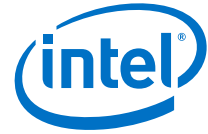
1.2.4 OpenOCD

- OpenOCD support is available with OpenOCD source
- GDB* server and Telnet* server support
- Halt/Step/Resume CPU
- CPU register access
- Memory access
- IO Access (via OpenOCD command tool, not via GDB)

1.3 Limitations

The software package has the following limitations:

- S3 support is implemented but not validated. It is not recommended for use in this release.
- Legacy SPI flash recovery not implemented.
- 1588 time-stamping protocol not supported in this release.
- Watchdog timer not enabled.
- Automatic version number updating during the update/recovery process is not implemented. Rollback protection (preventing downgrading to a previous software version) requires the version number of a software module to be greater or equal to the corresponding version number stored in the SPI flash. Support to update the version number stored in SPI flash if the corresponding software module is being updated, has not been added.
- UEFI 2.3.1 Secure Boot support is not implemented.
- eADC firmware update utility is not included in this release.
- Support for multiple keys is not included in this release.
- Backup image support is not included in this release.



- CAN limitations:
 - Bitrate limitation: 500 kbps is the highest speed supported on the current CAN channel. 10 kbps is not supported due to a hardware limitation of the MB91520 Fujitsu CAN controller.
 - Timestamps are not currently added by the IOC firmware prior to sending CAN messages over the UART to the Intel® Quark™ SoC X1000.
- UART DMA driver supports transmit only.
- UART DMA driver supports interrupt mode only.

1.4 Unplanned Functionality

Support for the following items is not plan of record (POR):

- Network boot
- Legacy OS boot
- ECC scrubbing (also called memory patrolling)

1.5 Component Versions

1.5.1 Packages

```
Board_Support_Package_Sources_for_Intel_Quark_v1.0.1:
grub-legacy_5775f32a+v1.0.1.tar.gz
meta-clanton_v1.0.1.tar.gz
Quark_EDKII_v1.0.1.tar.gz
quark_linux_v3.8.7+v1.0.1.tar.gz
sha1sum.txt
spi-flash-tools_v1.0.1.tar.gz
sysimage_v1.0.1.tar.gz

CAN_Firmware_for_Intel_Quark_v1.0.1.zip (selected customers only)
```

1.5.2 BIOS/Firmware Version

Development Platform	Version
Clanton Peak SVP	1.0

1.6 Related Documentation

The documents in [Table 1](#) provide more information about the software in this release.

Table 1. Related Documentation (Sheet 1 of 2)

Document Name	Reference Number
Intel® Quark™ SoC X1000 Software Release Notes (this document)	330232
Intel® Quark™ SoC X1000 Board Support Package (BSP) Build and Software User Guide	329687
Intel® Quark™ SoC X1000 Software Developer's Manual for Linux*	330235
Intel® Quark™ SoC X1000 Secure Boot Programmer's Reference Manual	330234

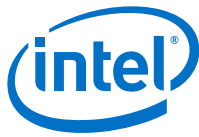


Table 1. Related Documentation (Sheet 2 of 2)

Document Name	Reference Number
Intel® Quark™ SoC X1000 UEFI Firmware Writer's Guide	330236
Intel® Galileo Board User Guide	330237
Source Level Debug using OpenOCD/GDB/Eclipse on Intel® Quark SoC X1000 Application Note https://communities.intel.com/docs/DOC-22203	330015
Intel® Quark™ SoC X1000 Datasheet https://communities.intel.com/docs/DOC-21828	329676
Intel® Quark™ SoC X1000 Core Developer's Manual https://communities.intel.com/docs/DOC-21826	329679
Intel® Quark™ SoC X1000 Core Hardware Reference Manual https://communities.intel.com/docs/DOC-21825	329678
Clanton Hill and CAN Getting Started Guide This document is provided to selected customers only; contact your Intel representative.	545350

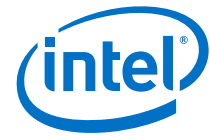
1.7 Licensing

This package contains source code licensed under one or more open source licenses. Consult the COPYING, README, or LICENSE files in the appropriate subdirectory. Intel does not make any representations or warranties, express or implied, including without limitation, any warranty of fitness for any purpose, merchantability or non-infringement.

The package also includes executable binaries provided under Intel Proprietary License (IPL) as listed in Table 2. The IPL license file is in the same directories as the binaries in the package.

Table 2. License Files

Location	Description
...\\Quark_EDKII_v1.0.1 \\QuarkSocPkg\\QuarkNorthCluster\\Binary\\QuarkMicrocode\\RMU.bin	Microcode for the Intel® Quark™ SoC X1000. (RMU: Remote Management Unit)
...\\Quark_EDKII_v1.0.1 \\QuarkSocPkg\\QuarkNorthCluster\\Binary\\Quark2Microcode\\RMU.bin	Microcode for a future generation Quark SoC.
...\\CAN_Firmware_for_Intel_Quark_v1.0.1 \\ioc_combined_image_clanton.mhx	CAN firmware to be programmed to the Fujitsu CAN controller on a Clanton Hill FFRD.

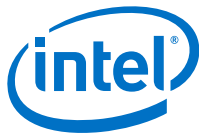


2.0 Known Issues

Known issues in the current release are listed below.

Table 3. Known Issue Summary

38292 - Cannot force MMC into 4-bit mode due to kernel bug.....	10
45539 - SDMediaDevice.efi is setting older 25 MHz cards to 50 MHz.....	10
46834 - UART interrupt handler not restored after resume from S3.....	10
48226 - eSRAM driver cannot map code required to do mapping.....	10
53887 - Deadlock in bluetooth stack - inherited from upstream kernel.....	11
57071 - Galileo board is unavailable after host computer sleeps.....	11
58381 - Attempting to unload a Linux driver which is in use causes console to freeze.....	11
58453 - pch_udc driver crash on reload.....	11
60003 - Legacy RTC 'Valid' time bit is set even though RTC contains invalid time.....	12
60147 - Quark enumerates incorrect device class as a USB CDC ACM device.....	12
60803 - BIOS error when using 2G MMC card.....	12
61236 - Real Time Clock update issue (sh: %4Y%2m%2d%2H%2M: bad number).....	12
63520 - SMBIOS fields are currently incorrect for the Quark reference platforms.....	13
64225 - SPI flash tool does not detect duplicate sections or duplicate options in a section in layout.conf ...	13
64263 - Error detecting Western Digital USB 3.0 hard drive.....	13
64428 - Legacy Resume Well GPIO registers showing hardware default values after cold boot on Clanton Hill board.....	13
65706 - Hot plug of USB key intermittently fails.....	14
65952 - USB Errors seen with Sandisk Cruzer 4GB Flash Drive.....	14
66053 - Poor USB write performance caused by automounter.....	14
66218 - Nonfunctional USB key may break the detection for other functional USB keys on Clanton Hill.....	14
66803 - Recovery boot intermittently stalls during PCI enumeration.....	15
69965 - Quark EDKII default exception handler entry point is not valid.....	15
70897 - SPI flash corruption can cause a system built without the SECURE_LD build option to become unrecoverable.....	15
70961 - Clanton Hill: If ETH0 is disconnected, ETH1 will not automatically pick up an address from DHCP .	15
73738 - CrossHill: cannot access USB stick if you boot Linux from SD card or USB stick when both devices plugged in during power on.....	15
73848 - Spurious 'unmounting /media/realroot' error message.....	16
74444 - No value returned in EventLogLastEntry output parameter in EFI_TCG_HASH_LOG_EXTEND_EVENT Service of EFI_TCG_PROTOCOL.....	16
75161 - Boot log error: memory range cannot be reserved.....	16
75172 - Clanton Hill: USB Error messages reported when booting debug build of EDKII.....	16
75539 - Legacy GPIO driver does not detect multiple, synchronous interrupts.....	17
77401 - Clanton Hill board hangs after checking or setting speed of ttyQRK0 (stty).....	17
77507 - Galileo Gen2 only: IRQs missed if pulses too close together on GPIO expanders.....	17
77674 - Parity error checking is performed even when set to ignore errors.....	18
77914 - UART DMA: incrementation of an array in an ifdef statement causes driver to crash.....	18
78401 - Sketch performance impacted when USB serial cable is removed.....	18



78550 - Some USB keys not recognised by Quark EDKII recovery on Galileo and Galileo Gen2..... 19
78738 - I2C/GPIO level-triggered interrupts cause system hang 19
79563 - Galileo USB device not recovering on reinsert 19

2.1 38292 - Cannot force MMC into 4-bit mode due to kernel bug

Title	Cannot force MMC into 4-bit mode due to kernel bug
Id	38292
Implication	There is a kernel bug that is seen when forcing MMC into 4-bit mode. If you use the command: modprobe sdhci debug_quirks=0x400000 Only one bit is set: SDHCI QUIRK_FORCE_1_BIT_DATA, bit 22 The board fails to initialize; returning these errors: - 110 timeout - 5 I/O error
Workaround	Use the command: modprobe sdhci debug_quirks=0x8400000 This sets: SDHCI QUIRK_FORCE_1_BIT_DATA, bit 22 SDHCI QUIRK_MISSING_CAPS, bit 27

2.2 45539 - SDMediaDevice.efi is setting older 25 MHz cards to 50 MHz

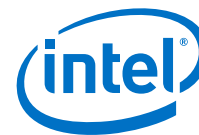
Title	SDMediaDevice.efi is setting older 25 MHz cards to 50 MHz
Id	45539
Implication	25MHz SD cards will not be recognized or usable.
Workaround	Use 'Fast' 50MHz capable SD cards.

2.3 46834 - UART interrupt handler not restored after resume from S3

Title	UART interrupt handler not restored after resume from S3
Id	46834
Implication	Suspected race condition between 8250 restore code and interrupt handler. Following resume from S3, 8250 will be in polled mode, not interrupt mode.
Workaround	Do not enter into S3 (unsupported).

2.4 48226 - eSRAM driver cannot map code required to do mapping (Sheet 1 of 2)

Title	eSRAM driver cannot map code required to do mapping
Id	48226



2.4 48226 - eSRAM driver cannot map code required to do mapping (Sheet 2 of 2)

Implication	eSRAM driver depends on code internally and externally in order to map things into eSRAM. During the mapping process, over-layed sections of DRAM become NULL for a time. It is not possible to eSRAM overlay code to itself be overlaid.
Workaround	Do not try to overlay any of the following kernel symbols: intel_cln_esram_* intel_cln_sb_* memcpy spin_lock spin_unlock spin_lock_irqsave spin_unlock_irqrestore pci_read_config_dword pci_write_config_dword

2.5 53887 - Deadlock in bluetooth stack - inherited from upstream kernel

Title	Deadlock in bluetooth stack - inherited from upstream kernel
Id	53887
Implication	When using the bluetooth software stack, a potential deadlock message can be found in /var/log/messages. Could potentially cause a lock-up but this has yet to be shown.
Workaround	None.

2.6 57071 - Galileo board is unavailable after host computer sleeps

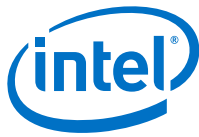
Title	Galileo board is unavailable after host computer sleeps
Id	57071
Implication	When the Galileo board is connected to a host computer that enters sleep mode, and the host is woken, the Galileo board will be unavailable on USB. This behavior is caused by the Gadget Serial driver and is seen on all OSes (Linux, Windows, Mac OS).
Workaround	There is no workaround, you must reboot the Galileo board.

2.7 58381 - Attempting to unload a Linux driver which is in use causes console to freeze

Title	Attempting to unload a Linux driver which is in use causes console to freeze
Id	58381
Implication	When a driver is in use (like for instance SD/MMC mass storage device when an SD card is mounted) and user tries to remove it using 'modprobe -r mmc_block' then existing console hangs. Existing console is not usable until board rebooted or mass storage device unmounted from other console.
Workaround	Make sure the driver is not use before trying to unload. For instance unmount mass storage device first, then unload mmc_block driver.

2.8 58453 - pch_udc driver crash on reload (Sheet 1 of 2)

Title	pch_udc driver crash on reload
Id	58453



2.8 58453 - pch_udc driver crash on reload (Sheet 2 of 2)

Implication	When ehci_pci, ehci_hcd, pch_udc, g_serial drivers are loaded and user executes: modprobe -r g_serial modprobe -r pch_udc modprobe pch_udc then pch_udc driver crashes. Problem seen on Galileo board. Driver is unusable until board rebooted.
Workaround	Unload first ehci-pci driver to revert to USB1.1, then g_serial and pch_udc drivers can be unloaded or reloaded.

2.9 60003 - Legacy RTC 'Valid' time bit is set even though RTC contains invalid time

Title	Legacy RTC 'Valid' time bit is set even though RTC contains invalid time
Id	60003
Implication	Legacy RTC 'Valid' time bit is set even though RTC contains invalid time. Any software that trusts the 'Valid' bit without any sanity checks on the time/date may be using a corrupt date/time.
Workaround	None.

2.10 60147 - Quark enumerates incorrect device class as a USB CDC ACM device

Title	Quark enumerates incorrect device class as a USB CDC ACM device
Id	60147
Implication	As a USB CDC ACM device, the Quark SoC enumerates a USB Device descriptor with Class, SubClass and DeviceProtocol 02, 00, and 00 respectively. This is incorrect given that the Quark CDC ACM setup uses Interface Association Descriptors. The USB specification recommends different values in the device descriptor when using IADs, consequently, Windows may generate errors. The values in the device descriptor should be EFh, 02h, 11h, respectively.
Workaround	None.

2.11 60803 - BIOS error when using 2G MMC card

Title	BIOS error when using 2G MMC card
Id	60803
Implication	2G Transcend MMC card (TS2GMMC4) is not recognised or is unusable.
Workaround	Use alternative MMC card.

2.12 61236 - Real Time Clock update issue (sh: %4Y%2m%2d%2H%2M: bad number) (Sheet 1 of 2)

Title	Real Time Clock update issue (sh: %4Y%2m%2d%2H%2M: bad number)
Id	61236



2.12 61236 - Real Time Clock update issue (sh: %4Y%2m%2d%2H%2M: bad number) (Sheet 2 of 2)

Implication	The initscripts provided in poky release 1.4 do not support the simplified date program used by busybox. This shows an error in the boot log and may prevent Linux from reading time from the RTC clock and from saving time to it.
Workaround	Go to the /etc/init.d/ directory on the target system. In both the bootmisc.sh and save-rtc.sh scripts there, search for: date -u +%4Y%2m%2d%2H%2M and replace with: date -u +%Y%m%d%H%M

2.13 63520 - SMBIOS fields are currently incorrect for the Quark reference platforms

Title	SMBIOS fields are currently incorrect for the Quark reference platforms
Id	63520
Implication	Only SMBIOS Type0 and Type2 fields have been validated to be correct. Software using any other SMBIOS entries may be using incorrect information.
Workaround	Only use validated SMIOS table entries.

2.14 64225 - SPI flash tool does not detect duplicate sections or duplicate options in a section in layout.conf

Title	SPI flash tool does not detect duplicate sections or duplicate options in a section in layout.conf
Id	64225
Implication	The python module ConfigParser used by the SPI flash tools cannot process multiple sections in the layout.conf file that do not have unique names. Within each section, it cannot process options that are not unique. This behavior may cause unexpected results. For example, if the layout.conf file has two sections called [Main], the second section will be used. If the layout.conf file has two options both called Size in the [Main] section, the second Size option will be used.
Workaround	Do not duplicate section names or option names within the same section in layout.conf.

2.15 64263 - Error detecting Western Digital USB 3.0 hard drive

Title	Error detecting Western Digital USB 3.0 hard drive
Id	64263
Implication	Western Digital USB3.0 HDD not recognized or usable.
Workaround	Use alternative USB HDD.

2.16 64428 - Legacy Resume Well GPIO registers showing hardware default values after cold boot on Clanton Hill board (Sheet 1 of 2)

Title	Legacy Resume Well GPIO registers showing hardware default values after cold boot on Clanton Hill board
Id	64428



2.16 **64428 - Legacy Resume Well GPIO registers showing hardware default values after cold boot on Clanton Hill board (Sheet 2 of 2)**

Implication	During Automating testing, certain Quark SoC Legacy Bridge Resume Well GPIO registers have shown hardware defaults after system cold boot. These registers include: Resume Well GPIO Input/Output Select (RGIO)—Offset 24h Resume Well GPIO Trigger Negative Edge Enable (RGTNE)—Offset 30h Resume Well GPIO GPE Enable (RGGPE)—Offset 34h Software and hardware components dependent on SoC resume well GPIOs may fail. This includes Battery Charge Enable Output, Main Battery Valid Input, Accelerometer Wake Input, PCIe reset output, WiFi disable output and GPS Antenna Enable Output. Registers have always shown correct values after a system warm boot.
Workaround	None.

2.17 **65706 - Hot plug of USB key intermittently fails**

Title	Hot plug of USB key intermittently fails
Id	65706
Implication	USB key is not recognized or is unusable.
Workaround	Disconnect and reconnect the USB key.

2.18 **65952 - USB Errors seen with Sandisk Cruzer 4GB Flash Drive**

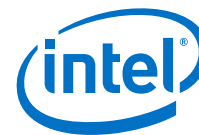
Title	USB Errors seen with Sandisk Cruzer 4GB Flash Drive
Id	65952
Implication	USB Key 'Sandisk Cruzer 4GB' is not recognized or is unusable in BIOS.
Workaround	Use alternative USB key.

2.19 **66053 - Poor USB write performance caused by automounter**

Title	Poor USB write performance caused by automounter
Id	66053
Implication	Automounting of USB memory is done with the '-o sync' flag by default. For VFAT filesystems (the default on USB and SD memory), there is a performance degradation which causes a typical write to take about 5 minutes.
Workaround	One workaround is to search and replace '-o sync' with '-o flush' in the /usr/bin/automount.sh file. However, the copy command will return before the write is complete. If the USB memory device is removed before the write is complete, the board may be in an unbootable state.

2.20 **66218 - Nonfunctional USB key may break the detection for other functional USB keys on Clanton Hill**

Title	Nonfunctional USB key may break the detection for other functional USB keys on Clanton Hill
Id	66218
Implication	This issue is seen only when non-functional USB key is connected to J1. Note that J12 (USB port0) and functional USB key connected to J10 (USB port1 via hub). Issue is not seen when positions are swapped.
Workaround	Only connect functional USB devices (USB devices that EDKII can function with without errors) to the system.



2.21 66803 - Recovery boot intermittently stalls during PCI enumeration

Title	Recovery boot intermittently stalls during PCI enumeration
Id	66803
Implication	An intermittent system hang has been observed when booting a recovery image. This hang occurs during PCI enumeration. This hang has only been observed on a Cross Hill platform and happened 4 times out of 10 attempts.
Workaround	Retry the recovery process.

2.22 69965 - Quark EDKII default exception handler entry point is not valid

Title	Quark EDKII default exception handler entry point is not valid
Id	69965
Implication	If the system hits an exception (divide by zero for example) during Quark EDKII boot then the system will vector to the default exception handler at address 0xFFFFFE4. As there is no valid exception handler at this address, system behavior is undefined.
Workaround	None.

2.23 70897 - SPI flash corruption can cause a system built without the SECURE_LD build option to become un-recoverable

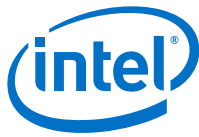
Title	SPI flash corruption can cause a system built without the SECURE_LD build option to become un-recoverable
Id	70897
Implication	The RMU.bin area of SPI flash is not currently protected by the Protected BIOS Range Registers (RCBA + 3080h -> RCBA + 308Bh). The RMU.bin is a critical component that is required for the Recovery boot path. If the RMU.bin image in SPI flash gets corrupted during SPI flash updates, then the system will be unrecoverable and unable to boot.
Workaround	Avoid updating this area of SPI flash on unsecure systems.

2.24 70961 - Clanton Hill: If ETH0 is disconnected, ETH1 will not automatically pick up an address from DHCP

Title	Clanton Hill: If ETH0 is disconnected, ETH1 will not automatically pick up an address from DHCP
Id	70961
Implication	There are two PHYs on the Clanton Hill board. If ETH0 is disconnected and ETH1 is connected to the network with DHCP available, an address for ETH1 is not retrieved automatically.
Workaround	Enter the command 'ifup ETH1' to manually retrieve an address.

2.25 73738 - CrossHill: cannot access USB stick if you boot Linux from SD card or USB stick when both devices plugged in during power on

Title	CrossHill: cannot access USB stick if you boot Linux from SD card or USB stick when both devices plugged in during power on
Id	73738
Implication	Issue seen on Cross Hill platforms only. If both SD card and USB key are connected to the board during power on, the USB key cannot be accessed.
Workaround	Insert removable device later after boot.



2.26 73848 - Spurious 'unmounting /media/realroot' error message

Title	Spurious 'unmounting /media/realroot' error message
Id	73848
Implication	When booting from mass storage the following error is returned in the boot log: umount: can't umount /media/realroot: Device or resource busy This occurs for images booted from mass storage devices.
Workaround	None; this error can be ignored.

2.27 74444 - No value returned in EventLogLastEntry output parameter in EFI_TCG_HASH_LOG_EXTEND_EVENT Service of EFI_TCG_PROTOCOL

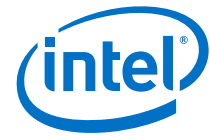
Title	No value returned in EventLogLastEntry output parameter in EFI_TCG_HASH_LOG_EXTEND_EVENT Service of EFI_TCG_PROTOCOL
Id	74444
Implication	UEFI Applications / Bootloaders using the EFI_TCG_PROTOCOL installed by software release 1.0.1 will not receive any value in the EventLogLastEntry output parameter in EFI_TCG_HASH_LOG_EXTEND_EVENT Service of EFI_TCG_PROTOCOL. This is a known issue with -r13937 of the EDKII SecurityPkg.
Workaround	Do not use value in EventLogLastEntry after calling EFI_TCG_HASH_LOG_EXTEND_EVENT Service of EFI_TCG_PROTOCOL.

2.28 75161 - Boot log error: memory range cannot be reserved

Title	Boot log error: memory range cannot be reserved
Id	75161
Implication	When booting, the following error is displayed in boot logs: [0.996963] pnp: PnP ACPI init [0.996963] ACPI: bus type pnp registered [1.003633] system 00:00: [mem 0xe0000000-0xe1ffffff] has been reserved [1.011283] system 00:00: [mem 0xfed1c000-0xfed1ffff] has been reserved [1.018649] system 00:00: [mem 0x000c0000-0x000dffff] has been reserved [1.026093] system 00:00: [mem 0x000e0000-0x000fffff] could not be reserved
Workaround	This error message will not affect board operation and can be ignored.

2.29 75172 - Clanton Hill: USB Error messages reported when booting debug build of EDKII

Title	Clanton Hill: USB Error messages reported when booting debug build of EDKII
Id	75172
Implication	The following error messages are reported during boot on Clanton Hill with a debug build of EDKII: Error Count : 3 EhcControlTransfer: error - Device Error, transfer - 2 However, no functional USB issues are observed and USB is working as expected. Issue is currently under investigation.
Workaround	None.



2.30 75539 - Legacy GPIO driver does not detect multiple, synchronous interrupts

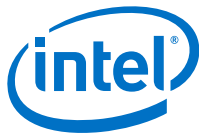
Title	Legacy GPIO driver does not detect multiple, synchronous interrupts
Id	75539
Implication	<p>Error was observed during testing on the Legacy GPIO, when interrupts are generated at the same time for multiple pins. The setup is as follows: One GPIO pin is set as an output (GPIO_X), two are set as inputs (GPIO_Y, GPIO_Z). All pins are connected together. GPIO_Y and GPIO_Z are set to interrupt on a rising edge. GPIO_X is set to 1.</p> <p>The output behavior is as follows: GPIO_Y interrupt count increases by one in /proc/interrupts. GPIO_Z interrupt count does not change in /proc/interrupts.</p> <p>This behavior is not observed in the I2C/GPIO driver.</p>
Workaround	This can be fixed by following the method used in the I2C/GPIO driver. First the register is read only once and the mask is saved locally. All bits are then cleared together and each interrupt is addressed using the local mask.

2.31 77401 - Clanton Hill board hangs after checking or setting speed of ttyQRK0 (stty)

Title	Clanton Hill board hangs after checking or setting speed of ttyQRK0 (stty)
Id	77401
Implication	<p>On the Clanton Hill board when trying to use stty, sometimes the command hangs and nothing happens. This is likely due to the port being stuck because it's waiting for one of the modem control lines to be asserted.</p> <p>When the CAN microcontroller on the Clanton Hill board is reset, it sends some data over UART. Then when the stty command is run, the system gets stuck.</p>
Workaround	Reset the Fujitsu CAN microcontroller again to restore the system.

2.32 77507 - Galileo Gen2 only: IRQs missed if pulses too close together on GPIO expanders (Sheet 1 of 2)

Title	Galileo Gen2 only: IRQs missed if pulses too close together on GPIO expanders
Id	77507



2.32 77507 - Galileo Gen2 only: IRQs missed if pulses too close together on GPIO expanders (Sheet 2 of 2)

Implication	<p>On the Galileo Gen2 boards, there is a PCAL9555A GPIO expander that provides interrupt support for some of the digital I/O header pins. Those pins are IO2-3 and IO14-9, and also the shield reset button.</p> <p>If those pins are configured to generate interrupt notifications, and if the rate of interrupt trigger events (e.g. falling/rising edge signals on the pin) exceeds a combined rate of approximately 1000 interrupt events per second, the interrupt notifications from the PCAL9555A may stop working. Notifications for subsequent interrupt events will not be received by software.</p> <p>Possible implications:</p> <ul style="list-style-type: none"> * 'Change-mode' interrupts (where an interrupt is generated on either a rising or a falling edge input signal) should be used on IO2-3 only if the rate of interrupts is likely to exceed 1000 per second. For other interrupt modes (falling edge only, rising edge only, low level, high level), it is possible to use SoC GPIO pins instead which are also connected to IO2-3. * Interrupts should be used on IO14-19 only if the rate of interrupts is likely to exceed 1000 per second. However, due to the presence of 1uF capacitors on these pins and their effect on signal rise/fall times, it is unlikely that these pins would be used for high-rate interrupt signalling. * The shield reset input is intended for use with a manually-pressed reset button on an Arduino shield. In that scenario, the rate of button presses is unlikely to exceed 1000 per second. However, there is a chance that signal bounce from the mechanical switch could conceivably trigger this scenario.
Workaround	<p>It is possible to restore interrupt functionality by reading the current input values from any GPIO pin(s) on the PCAL9555A that are configured to generate interrupts. This will effectively 'clear' the outstanding interrupts and allow new interrupt notifications to be detected by software.</p>

2.33 77674 - Parity error checking is performed even when set to ignore errors

Title	Parity error checking is performed even when set to ignore errors
Id	77674
Implication	If parity checking is set on Quark via stty with the ignpar setting (ignore incoming packages with parity error), incoming packages with parity error are not ignored as expected.
Workaround	Handle parity errors at an application level instead of depending on the stty ignpar setting.

2.34 77914 - UART DMA: incrementation of an array in an ifdef statement causes driver to crash

Title	UART DMA: incrementation of an array in an ifdef statement causes driver to crash
Id	77914
Implication	Removing CONFIG_SERIAL_QUARK_UART_CONSOLE from the kconfig causes an array in the UART driver to not increment. As a result, both ports are not properly addressed, which then causes the driver to crash.
Workaround	Do not build the kernel without CONFIG_SERIAL_QUARK_UART_CONSOLE

2.35 78401 - Sketch performance impacted when USB serial cable is removed (Sheet 1 of 2)

Title	Sketch performance impacted when USB serial cable is removed
Id	78401



2.35 78401 - Sketch performance impacted when USB serial cable is removed (Sheet 2 of 2)

Implication	After a sketch has been downloaded to the board, and the USB cable that was used to download the sketch is removed, the LED blinking slows and becomes erratic. Also, cloader is stuck with stale file handles and the console cannot be used.
Workaround	This is a known issue with the cloader and related to other open issues on the gadget-serial interface. High performance sketches are more affected than lower ones. This issue was seen when designing sketches that will operate without USB cable.

2.36 78550 - Some USB keys not recognised by Quark EDKII recovery on Galileo and Galileo Gen2

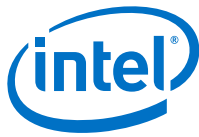
Title	Some USB keys not recognised by Quark EDKII recovery on Galileo and Galileo Gen2
Id	78550
Implication	Recovery process will fail on Galileo and Galileo Gen2 with these USB keys. Currently the following USB keys have been seen to fail: 1) Sandisk cruzer 4GB 2) Transend 4GB
Workaround	Two potential workarounds have been identified: (1) Connect a USB hub to the Galileo Gen2 USB port and then connect the failing USB key(s) to the USB hub. The USB keys have been observed to pass in this configuration (2) Select a different USB key

2.37 78738 - I2C/GPIO level-triggered interrupts cause system hang

Title	I2C/GPIO level-triggered interrupts cause system hang
Id	78738
Implication	System hangs during testing level-triggered interrupt handling in the I2C/GPIO driver (intel_qrk_gip). After loading the driver, the GPIO pin level goes low (verified with multimeter) and stays low. The interrupt fires and the system hangs forever (no response on shell via serial or ssh).
Workaround	Reboot the board.

2.38 79563 - Galileo USB device not recovering on reinsert

Title	Galileo USB device not recovering on reinsert
Id	79563
Implication	Attempts to reconnect USB cable to the device port result in device not recognized. Galileo Gen2 (and FabD) appear to be unable to complete their initialization events after a cable reinsertion. There may also be a dependence on particular electrical characteristics of the cable termination connection, which may be improved / altered by different cables or switches. Connection of extra test equipment (USB protocol analyzer from LeCroy) showed no failures. It is possible that this piece of equipment put the USB cable in a good state again. The issue has been reproduced on Cross Hill boards running Yocto (3.8.7) and WR IDP (3.4).
Workaround	Stop all sketches via console port and reload the g_serial. Connecting via a powered USB hub or disconnecting only from the host end of the cable appears to reduce occurrences of the problem.



3.0 Resolved Issues

This section contains issues resolved since package version 1.0.0.

Table 4. Resolved Issue Summary

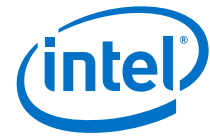
38542 - SPI flash tool / signing tools does not support multiple inclusions of same binary at different addresses.....	20
71061 - Linux boot failure on failure to remap PCIe MMIO region (256MB) from physical to virtual addressing.....	20
71538 - Linux segfault when using lock prefix instruction under specific circumstances	21
73384 - IRQ unhandled exception occurs when running sketch	21
74073 - System hangs before system bootloader / payload is executed	21
75904 - OpenSSL version affected by 'heartbleed' defect	22

3.1 38542 - SPI flash tool / signing tools does not support multiple inclusions of same binary at different addresses

Title	SPI flash tool / signing tools does not support multiple inclusions of same binary at different addresses
Id	38542
Implication	<p>When building an image using a layout.conf file that uses the same 'item_file' source in two (or more) asset descriptor blocks, the expected behavior is as follows:</p> <ol style="list-style-type: none"> 1. Image is generated. 2. Two assets exist at different locations, with identical body data. 3. Even though the bodies of both assets contain identical data, the RSA signature section of each asset should contain different signatures, due to the intentionally non-deterministic nature of the signing process. <p>What actually happens:</p> <p>All assets will be duplicates of the last asset listed in layout.conf, including RSA signatures and any other variables such as SVN indices.</p> <p>If, for example, 3 assets use the same 'item_file' source, and have SVN indices of 1, 2, and 3 respectively in layout.conf, and the one with SVN index 3 is the last one listed in layout.conf, then the other two assets that use this same 'item_file' source will also have an SVN index of 3, as well as identical RSA signatures.</p>
Resolution	Resolved in release 1.0.1.

3.2 71061 - Linux boot failure on failure to remap PCIe MMIO region (256MB) from physical to virtual addressing (Sheet 1 of 2)

Title	Linux boot failure on failure to remap PCIe MMIO region (256MB) from physical to virtual addressing
Id	71061



3.2 71061 - Linux boot failure on failure to remap PCIe MMIO region (256MB) from physical to virtual addressing (Sheet 2 of 2)

Implication	V1.0.0 firmware required the operating system to map PCI express MMIO space from physical to virtual address. However, in the 1.0.0 release, the kernel called UEFI runtime service SetVirtualAddressMap() without PCI express MMIO space being mapped to virtual addresses. The impact was the system would reboot in Recovery mode earlier in kernel boot.
Resolution	Resolved in release 1.0.1.

3.3 71538 - Linux segfault when using lock prefix instruction under specific circumstances

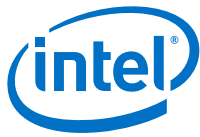
Title	Linux segfault when using lock prefix instruction under specific circumstances
Id	71538
Implication	When a memory instruction with LOCK prefix executes and if it encounters a page fault (#PF), the state of the CPU could potentially get corrupted. Software should avoid using the LOCK prefix for instructions that may cause page fault (#PF).
Resolution	<p>Resolved in release 1.0.1.</p> <p>Due to the LOCK prefix core silicon errata, the Yocto software release has patched the GNU assembler to remove LOCK instructions from code generated by the GNU toolchain. The workaround is enabled by default and no option has to be specified. All code is compiled with the workaround applied, so no binaries or libraries will include the LOCK prefix.</p> <p>The toolchain workaround can be verified to be in the toolchain by issuing the GNU assembler command: <code>> as --help</code></p> <p>The help text will show the option: <code>-mquark-strip-lock=[yes no] strip all lock prefixes; default is yes</code></p> <p>The workaround can be explicitly set/cleared from gcc compiler using the command: <code>gcc -Xassembler -mquark-strip-lock=[yes no]</code></p>

3.4 73384 - IRQ unhandled exception occurs when running sketch

Title	IRQ unhandled exception occurs when running sketch
Id	73384
Implication	When running sketch on Galileo board, an IRQ error occurs. Sometimes it stops with IRQ40, other times the sketch simply stops executing, leaving GPIOs stuck in whatever state they were in. The last instance, the LED was frozen on my number counter (1 digit lit). Root was available on serial console and no errors or messages in the system log.
Resolution	Resolved in release 1.0.1. Modified gadget driver (udc_pch.c) with additional logic to the pch_udc ISR for IRQ_NONE conditions to cater for valid handled interrupts (IRQ_HANDLED).

3.5 74073 - System hangs before system bootloader / payload is executed

Title	System hangs before system bootloader / payload is executed
Id	74073
Implication	If the EDKII "MemoryConfig" boot services variable is corrupted or overridden, the system bootloader / payload will not be executed. On Quark base SKU systems, the user will also notice that EDKII boot menu will not be displayed.
Resolution	Resolved in release 1.0.1.



3.6 75904 - OpenSSL version affected by 'heartbleed' defect

Title	OpenSSL version affected by 'heartbleed' defect
Id	75904
Implication	A missing bounds check in the handling of the TLS heartbeat extension can be used to reveal up to 64k of memory to a connected client or server. Documented here: https://www.openssl.org/news/secadv_20140407.txt
Resolution	Resolved in release 1.0.1. The OpenSSL recipe was updated to download and build a fixed version of OpenSSL.

§ §