==========
Intel(R) Boot Loader Development Kit (Intel(R) BLDK) Core 2.3.6.8
[Based on the UEFI Development Kit 2010 (UDK2010)]
Crown Bay Release Notes
December 30, 2011
==========

==========

==========
CONTENTS
==========

==========
1. OVERVIEW
==========
This Intel(R) Boot Loader Development Kit (Intel(R) BLDK) is specifically
produced for the Crown Bay platform. The components that make up the kit are:

- The Intel(R) BLDK Development Application.
- The Crown Bay platform code base.
- The Intel(R) UEFI Development Kit Debugger Tool.
- Documentation including the Getting Started Guide and the User's Guide.

==========
2. RELEASE INFORMATION

==========
This is the 2.3.6.8 release of the software for the Crown Bay platform. It is based on the Intel UEFI Development Kit (UDK) 2010 Update3 (UP3) which is open source posted on TianoCore.org plus platform specific code which has been added to this release package. All required product features have been implemented; however, optimization for boot time performance is left to customer's porting efforts.

EDKII documentation can be downloaded from http://sourceforge.net/projects/edk2/files/.

General Documentation: EDKII User Manual and EDK II Module Writer's Guide.
Specifications: EDKII Build, FDF, INF, DSC, DEC and VFR Specification.

==========
3. SUPPORTED FEATURES
==========
The following features are supported and have been validated.

- CPU, Memory, Basic I/O Initialization
- Boot from SATA, SD, USB
- Feature configuration through the Intel(R) BLDK Development Application
- Windows* Tool Chain (Windows Source Package)
- GCC Tool Chain (Linux* Source Package)
- UEFI Specification 2.3
- Fast Boot in less than 3 seconds
- ACPI 3.0
- Intel(R) UDK Debugger Tool
- SPI Flash Update
- HD Audio
- Enhanced Intel SpeedStep(R) Technology
- Intel(R) Hyper-Threading Technology
- Processor Virtualization (VT-x)

==========
4. HARDWARE CONFIGURATION
==========
This release has booted to the operating systems listed in Section 8 on a Crown Bay customer reference board using the following configuration:

- PS/2 keyboard and mouse
- Serial console and LVDS console
- SATA hard disk (AHCI mode)
- USB mouse and keyboard
- USB thumb drive

- SD card

Crown Bay has memory down rather than DIMMs, therefore serial presence detect (SPD) is not utilized.

The latest Chipset Microcode (CMC) Rev. 2.1 (C0_22211.BIN) supports A0, B0, and B1. If another version of the CMC is desired, place the .bin file in the appropriate TCMicrocode directory and change the .fdf file to include that CMC file in the build.

==========
5. BUILD ENVIRONMENT OPERATING SYSTEM
==========

All validation of the Intel(R) Boot Loader Development Kit was done on Windows XP 32-Bit with Microsoft Visual Studio 2005, Windows 7 64-Bit with Microsoft* Visual Studio 2008 and Timesys* Fedora* Remix 14 with Linux GCC (4.5.1) Tool Chain.

These are the only supported combinations for Intel(R) BLDK.

==========
6. BUILD TOOLS REQUIRED on HOST MACHINE
==========

A. Windows Environment
  a. Microsoft Visual Studio 2005 or 2008.
  b. Intel ACPI Source Language (iASL) Compiler, version 20070508
    Note: Download "iasl-win-20070508.zip" from
      "http://www.acpica.org/downloads/Version_20070508.php"
      and extract file "iasl.exe" to C:\ASL
  c. Windows Driver Development Kit (WinDDK), version 3790.1830
    Note: Download WinDDK.3790.1830 from:
      "http://download.microsoft.com/download/9/0/f/
      90f019ac-8243-48d3-91cf-81fc4093ecfd/1830_usa_ddk.iso"
      Extract the DDK under "C:\1830_usa_ddk".
      Run "C:\1830_usa_ddk\x86\KitSetup.exe".
      This will install WinDDK.3790.1830 under "C:\WINDDK\3790.1830"
  d. Intel(R) BLDK Development Application, version 2.0.1
    Note: Download from "http://edc.intel.com/Software/
      Intel-Boot-Loader-Development-Kit/"
      Double-click on the .exe file to install.

B. Linux Environment (Timesys Fedora Remix 14)

a. GCC Tool Chain, version 4.5.1 20100924
Note: Type "yum groupinstall development-tools" in Terminal with root user.
b. Intel ACPI Source Language (iASL) Compiler, version 20100528
Note: Type "yum install iasl libuuid-devel" in Terminal with root user.
c. Intel(R) BLDK Development Application, version 2.0.0
Note: Download from "http://edc.intel.com/Software/
Intel-Boot-Loader-Development-Kit/"
Run the installation script as follows to install:
-$ chmod u+x install.sh
-$ ./install.sh

Refer to the Getting Started Guide and the User's Guide for further details.
Intel UEFI Development Kit Debugger Tool can be downloaded at
"http://www.intel.com/technology/efi/sw-debug.htm"

==========
7. FLASH IMAGE BUILD STEPS
==========

A. Windows Environment
a. Double click the CB-EDKII-PostGold-2.3.6.8.exe file and then
type in the destination directory.
b. Open Intel(R) BLDK 2.0.1 Development Application tool.
c. Create a new project. Refer to section 4.1 in the Getting Started Guide.
d. In the Development Application, add a debug build to the build options.
- File > Preferences > Build Tools.
- Add a "New Item" and name it "Build &Debug"
- In the Application text box, enter "nmake"
- In the Parameters text box, enter "debugbuild"
- Click Save
e. Under the Build menu, you will see "Build", "Build Debug", "Rebuild",
"Clean" and "Clean All" options.
- To generate a release build, click "Build".
- To generate a debug build, click "Build Debug".
- Click "Clean" to remove previous builds.
- Click "Rebuild" to clean and generate a release build.
f. After a successful build, CROWNBAY.FD is the filename of flash image which
is located under the build directory.

B. Linux Environment (Timesys Fedora Remix 14)
a. Copy the CrownBay release package to a workspace and extract the
release package at the worskspace by command:
- tar xzvf ../CB-EDKII-PostGold-2.3.6.8.tar.gz

b. Use either "-$ bldk" or "-$ /opt/intel/bldk/bin/bldk" command to open the Intel(R) BLDK 2.0.0 Development Application tool.
c. Create a new project. Refer to section 4.1 in the Getting Started Guide.
d. In the Development Application, add a debug build to the build options.
- File > Preferences > Build Tools.
- Add a "New Item" and name it "Build &Debug"
- In the Application text box, enter "make"
- In the Parameters text box, enter "debugbuild"
- Click Save
e. Under the Build menu, you will see "Build", "Build Debug", "Rebuild", "Clean" and "Clean All" options.
- To generate a release build, click "Build".
- To generate a debug build, click "Build Debug".
- Click "Clean" to remove previous builds.
- Click "Rebuild" to clean and generate a release build.
f. After a successful build, CROWNBAY.FD is the filename of flash image which is located under the build directory.

Note:
If build target or feature configuration has been changed, a clean build is required. (Perform "Clean" or "Clean All" in Build menu before generating a release/debug build, manually removing the "build" folder in source tree is not considered a clean operation)

Refer to the Getting Started Guide and the User's Guide for further details.

Note: Review the size of the CrownBay SPI Part, 2MB (0x0000000 – 0x01FFFFF).
For that, you have to adjust the starting address of the flash image:
- Debug image: Address (Hex): 0x0000000 - 0x01FFFFF
- Release image: Address (Hex): 0x0100000 - 0x01FFFFF

==========
8. OPERATING SYSTEMS BOOT SUPPORT
==========
Only "UEFI aware" target operating systems will boot using this version of BLDK. This means that operating systems and boot loaders that require BIOS INT calls will not boot. To boot with this version of BLDK, a modified version of elilo or grub2 will be required for either of the two supported operating systems (MeeGo* 1.2 and Timesys Fedora Remix 14) to boot.

For detailed instructions on making the required modifications and on getting supported operating systems to boot, refer to the white paper titled "Setting up OS environments using EDK-II based Intel(R) BLDK" available from EDC at the

link below.

Booting MeeGo 1.2 and Timesys Fedora Remix 14 on the following media have been validated:
- SATA hard drive
- SATA solid state drive
- USB drive
- SD card

Link to download:
- Timesys Fedora Remix 14: "https://linuxlink.timesys.com/intel/linux/"
- MeeGo 1.2: "https://meego.com/downloads"
- White paper "Setting up OS environments using EDK-II based Intel(R) BLDK": "http://www.intel.com/p/en_US/embedded/hwsw/software/bldk#download"

==========
9. LIMITATIONS
==========
- Only "UEFI aware" operating systems will boot using this version of BLDK. Refer to Section 8.
- When release package is extracted to a folder, the folder name is restricted to contain characters (a-zA-Z0-9_-.) only. Build failure will occur if other characters are used.
- Tool for Source Level Debug is not supported for Linux build.
- Support only EFI Shell 2.0 compatible application. EFI application that consumes old Shell interfaces won't be supported by EFI Shell 2.0. Workaround: Launch EFI Shell 1.0 image .efi on top of Shell 2.0 environment.
- TPM MOR and PP features are not enabled for certain platforms with SPI flash parts.
- SIO UART is needed to be used as console for OS boot if respective OS doesn't support IOH UART as console.
- Overwriting of "/opt/intel/udkdebugger/bin/udk-gdb-script" with the script provided in /CrownBayPlatformPkg/Application/UdkScript/" is needed to avoid TARGETS hangs when issuing "resettarget" in UDK debugger.

==========
10. KNOWN ISSUES
==========

Reference: 3658279

Title: FirmwareVendor PCD is incorrectly converted from Unicode to ASCII

Description: In MdeModulePkg.dec, PcdFirmwareVendor is defined as a null-terminated Unicode string. This PCD can be customized using Intel(R) Boot Loader Development Kit. Since, Intel(R) Boot Loader Development Kit updates this PCD as an ASCII string, this PCD is redefined as an ASCII string in the EDK-II based BLDK source code as well.

Implication: If an EFI application or a driver reads the PcdFirmwareVendor PCD directly, it should treat the PCD as a null terminated ASCII string.

Resolution: None

Workaround: 1. Convert the ASCII string to a Unicode string after reading the PCD.
2. Read the Firmware Vendor from the System Table and not from PCD.

Affected O/S: EFI Shell

Reference: 3658305

Title: Bad page kernel messages when booting Linux.

Description: When booting Linux with detailed kernel messages, the kernel will print out numerous messages saying "BUG: Bad page state..." follows by a call trace. This happens at numerous places in the boot log, but doesn't seem to be affecting functionality.

Implication: Bad page error messages are popped up inside TimeSys Fedora Remix 14 windows.

Resolution: None

Workaround:

Affected O/S: Timesys Fedora Remix 14

Reference: 3658339

Title: Shell time mismatch with OS time when the timezone in shell is changed.

Description: When the system clock is synced to hw clock, the hw clock displayed the exact date and time as system clock in OS. When the OS was powered off and a cold boot was performed into efi shell, the shell time was 8 hours less compared to the OS time (ie. sys clock). This happened even when timezone in shell 2.0 was set to +8:00.

Implication: EFI Shell time is mismatched with OS system time after EFI timezone is set and synced EFI Shell time with OS system time.

Resolution: None

Workaround: Use EFI Shell default timezone.

Affected O/S: EFI Shell

==========

11. IMPLEMENTATION NOTES

==========

- The customer is responsible for implementing security safeguards prior to productizing the Intel(R) Boot Loader Development Kit.