White Paper

Matthew Lee

Platform Application Engineer

Intel Corporation

Embedded Controller Usage in Low Power Embedded Designs

An Overview

September 2011

Executive Summary

Embedded Controllers are a common part of Intel's low power embedded reference designs and, therefore, an important consideration for system designers to create derivative designs for an ODM or OEM. This white paper will provide system designers with sufficient knowledge to understand the form, fit and function of Embedded Controllers and how best to include them in a design.

Contents

Background

Embedded Controllers (ECs) are often found in low power embedded reference designs, performing a range of Input/Output (I/O) and system management functions. While these ECs have been an understood and established part of Intel Architecture based laptop, netbook and now tablet designs, their purpose is not as well understood in the embedded design world.

Since embedded designs tend to utilize Super I/O ICs, programmable logic (such as CPLDs or FPGAs) or discrete logic rather than an embedded controller, it can be problematic to leverage a low power embedded reference design directly without knowing what the embedded controller is actually doing.

Solution

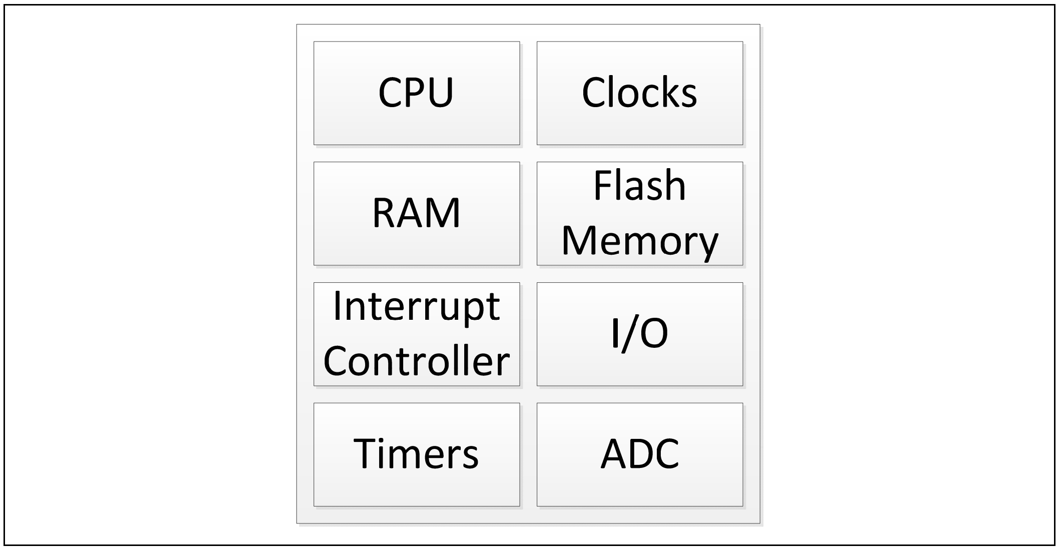
This white paper is intended to give readers an overview of Embedded Controller usage by covering the following subject areas:

- An overview of the function of an Embedded Controller
- The implementation of an Embedded Controller in low power embedded reference designs
- The ecosystem in place to service Embedded Controller implementations

What Is An Embedded Controller?

An Embedded Controller can be considered to be a microcontroller, with I/O and internal features targeted to suit the typical needs of a low power platform. With the microcontroller architecture in mind, Figure 1 summarizes the typical block diagram of a generic embedded controller.

Figure 1. Generic Embedded Controller Block Diagram



These blocks can be split into two main groups.

1. Hard-wired
2. Programmable

The hard-wired group comprises items which have one hardware defined function such as the clocks, interrupt controller, I/O, Timers and Analog-to-Digital Converter. The programmable group consists of the CPU, RAM & Flash memory blocks and the functions of those blocks are defined by the firmware programmed in the Flash memory.

Note: Different acronyms are often used instead of Embedded Controller and these describe subsets of the EC function.

- KSC: Keyboard and System Controller
- KBC: Keyboard Controller
- SMC: System Management Controller
- H8: A shortened name of the Embedded Controller family often use in reference designs: the Renesas* H8S Controller.

Embedded Controller Functional Overview

As discussed previously, the Embedded Controller consists of two groups: Hard-Wired and Programmable.

The hard-wired functionality is defined by the vendor of the Embedded Controller and can be further sub-divided into two groups: Keyboard Controller (KBC) and System Management Controller (SMC).

The programmable functionality is generally defined by the OEM/ODM that is developing the system. This programmability allows the Embedded Controller to be customized to the specific needs of the platform and to be differentiated from the same EC used by another OEM or ODM.

Hard-Wired Functionality: Keyboard Controller (KBC)

The keyboard controller aspects of Embedded Controller functionality often include the following items.

Intel 8042 Compatible Host Controller

This allows AT-compatible and PS/2-compatible support for keyboards and mice via reads & writes to I/O Ports 60h and 64h.

Legacy CPU Reset Control

This allows the CPU to be reset by asserting the RCIN# signal input to the platform ICH or PCH which, in turn, triggers the INIT# signal output from the ICH or PCH.

Legacy Fast A20 Gate Control

This allows the processor FSB address bit 20 (A20) to be forced low, ensuring DOS program compatibility, by asserting the A20GATE signal input to the platform ICH or PCH which, in turn, triggers the A20M# signal output from the ICH or PCH.

Note: Intel Architecture support for A20 gate support is gradually being removed in newer microarchitecture implementations.

Keyboard Matrix Scan Support

All keyboards have their keys arranged into a matrix of rows and columns. Due to number of signals needed to represent those row and columns, external keyboards use an onboard controller to translate from these row and column addresses to an electrically simpler protocol such as AT, PS/2 or USB.

In systems where the keyboard is integrated in the same chassis and the cable, is therefore, shorter or non-existent it can be simpler and more cost effective to dispense with the extra controller and connect directly to the system Keyboard Controller.

Shortcut Key Support

This allows the system to detect on certain key presses or combinations outside those keys normally supported by international keyboard standards. Such actions could include the following:

- Change screen brightness
- Enable or Disable wireless networking
- Increase or Decrease audio volume

Hard-Wired Functionality: System Management Controller (SMC)

The system management controller aspects of Embedded Controller functionality may include the following items.

Thermal Management

The Embedded Controller may have PWM interfaces that can be used to control system fans. The fans generally will be controlled using system input processed by the programmable functionality.

Power Monitoring

The Embedded Controller ADC signal can be used to monitor the voltage and, if using a sense resistor, the current consumption of specific power rails. This information could be used to, among other things, monitor battery charging or inform the user or administrator of potentially problematic power supply conditions.

Battery Management

In systems where batteries are one of the system power sources, the Embedded Controller can be used to control charging of the battery in addition to switching between the battery and AC adapter as the active power source changes or monitoring the various battery status metrics such as temperature, charge level and overall health.

SPI Flash Bridge

For older platforms, the Embedded Controller could be used to bridge between the ICH Firmware Hub (FWH) interface and the SPI interface, allowing the more common SPI flash to be used for BIOS storage rather than a native FWH.

ACPI Host Interface

The Embedded Controller may be capable of providing an ACPI compliant operating system (OS) with status and notifications regarding power management events. Further, it may also be capable of generating wake events to bring the system out of low power states.

Signal Buffering & Level Shifting

Some signals from the Intel Architecture components in the system may need to be buffered or level shifted in order to be utilized in the system. Some common examples are the buffering of the chipset SMBus, in order to drive more components than the chipset itself is capable of, and level shifting of the DDC signals from 3.3V to 5V for use at a VGA, DVI or HDMI connector. In order to reduce the BOM count and cost, the Embedded Controller can integrate these functions.

Programmable Functionality

The programmable aspects of the Embedded Controller are the primary method for differentiating the EC implementation for a specific design and/or OEM/ODM. In some cases, changes and additions to the hardware-wired functionality could be used to achieve a similar differentiation albeit with a higher investment most likely needed. In most cases, firmware based modifications are preferred since they are comparatively quick to implement and also to update or bug fix.

Examples of possible programmed functions follow.

Diagnostic Messages

The Embedded Controller can be used to detect system issues and communicate them either through the processor or directly to the user. Since the Embedded Controller is able to act independently of the Intel Architecture aspects of the system, it can be used to communicate diagnostic messages even when the IA aspects are not active such as before the CPU comes out of reset.

Docking Control

The Embedded Control can be used to detect the docking status of the platform, whether it is in a docking station or not. Based on this status, the EC can carry out tasks such as switching system power source to the dock, alter signal multiplexes to route signals from onboard interfaces to the dock and report the docking status to the operating system.

Power Plane Sequencing

In most low power platforms, there are many combinations in which the various power supplies can be implemented in both hardware device choice and schematic connection. The Embedded Controller firmware can be adjusted to suit the specific needs of the system whilst ensuring that power sequencing meets the needs of the processor and chipset.

This control is particularly critical for those platforms that integrate an Intel® Management Engine (Intel® ME). Historically, most systems had relatively simple single flows between different power states. In order to support technologies such as Intel® Active Management Technology (Intel® AMT), a larger set of flows have to be supported with a more granular control of power planes. For further information on ME requirements, each relevant platform should have the following document available for it: Intel® Management Engine (Intel® ME) and Embedded Controller Interaction for {Platform Codename} Platform Product Specification. Contact your Intel Field Sales Representative.

Thermal Management

The temperature of the processor, chipset, memory modules and other parts of the system can be actively monitored by reading thermal sensors, PECI transactions and other thermal reporting services. The system SMBus is often used to report at least some of the system thermal information. This information could be used to implement functions such as dynamic control of the system thermal solution and provision of user alerts.

PECI, in particular, will be an essential Embedded Controller functionality for some next generation platforms since it is planned to be the only method for thermal management of the processor's CPU and GFX cores.

Connecting Embedded Controller and Intel Architecture

In order for the Embedded Controller to work with the Intel Architecture components in a platform, some form of hardware link between the two is required. This link can be roughly split into five groups of signals, as described below.

Further information on these signals can be found in the various technical documents available for the Intel Architecture components of interest.

Low Pin Count (LPC) Bus

The Intel Architecture components in the system use the LPC bus to communicate with the Embedded Controller, and its functional blocks, as a peripheral device.

System Management Bus (SMBus)

The Intel Architecture components in the system can use the SMBus to communicate with the Embedded Controller, and its functional blocks, as a slave device. Additionally, if connected to the Intel Architecture SMLink interface, the Embedded Controller can use the SMBus to communicate with the Intel Architecture components, and the associated functional devices, as a slave device.

As previously stated, one of the uses of the SMBus communication is to transfer some system thermal information to the Embedded Controller.

Platform Environmental Control Interface (PECI)

As stated earlier in this document, the PECI link can be used to transfer thermal information from some Intel Architecture components to the Embedded Controller.

Power Management Sideband Signals

The Embedded Controller is in the center of power management for the platform so there are various power management signals connecting the Intel Architecture components to the Embedded Controller.

The functions of these signals can include:

- Indicating whether the system is running on AC or DC power supplies.
- System power button
- System reset button
- Power supply power good indicators
- Sx system state indicators
- Battery low indicator
- Platform reset indicator

Miscellaneous Sideband Signals

Other signals can also be connected to facilitate certain functionality. These signals can include:

- Interrupts
- Serial Peripheral Interface (SPI)
- A20 Gate Control
- Keyboard Controller Reset

Embedded Controllers in Low Power Reference Designs

As stated earlier in this document, many of Intel's low power embedded reference designs include an Embedded Controller. These Embedded Controllers implement many of the functions in the last chapter.

Hardware Support

Here is a list of Embedded Controllers that have been used in reference designs. Note that list is provided purely as a reference and does not constitute a specific recommendation.

Nuvoton* NPCE791E

- Intel® Atom™ Processor Z670 with Intel® SM35 Express Chipset Development Kit

Renesas* H8S/2161

- Mobile Intel® 915GME Express Chipset Development Kit

Renesas* H8S/2104

- Mobile Intel® 945GME Express Chipset Development Kit
- Intel® Core™2 Duo Processor T7500 and Mobile Intel® GME965 Express Chipset Development Kit

Renesas* H8S/2117

- Intel® Core™2 Duo Processor and Mobile Intel® 4 Series Express Chipset Family Development Kit
- Intel® Atom™ Processor Z5xx Series Development Kit
- 2nd Generation Intel® Core™ i7-2710QE and i5-2510E Processors with Intel® QM67 Express Chipset Development Kit
- Intel® Core™ i5-520E Processor with Mobile Intel® QM57 Express Chipset Development Kit

- Intel® Core™ i5-520M Processor with Mobile Intel® QM57 Express Chipset Development Kit
- Intel® Core™ i7-620LE Processor with Mobile Intel® QM57 Express Chipset Development Kit
- Intel® Core™ i7-620UE Processor with Mobile Intel® QM57 Express Chipset Development Kit

Firmware Support

Firmware for reference designs is provided by Intel in **binary form only**, to update the pre-flashed firmware. Source code for this firmware is not available due to the following general concerns:

- The code is subject to a number of 3rd party licenses
- The code contains protected IP from Intel, the Embedded Controller vendor and other hardware vendors.
- The code is not written or documented to a production-ready standard or feature set.

System designers requiring source code for Embedded Controller development should contact the 3rd parties listed in the next chapter.

Collateral Support

Collateral that documents the hard-wired functionality are available directly from the Embedded Controller vendors. There is no collateral that directly specifies the firmware-based Programmable functionality but there is some that covers aspects of the functionality

- The Intel® Management Engine (Intel® ME) and Embedded Controller Interaction for {Platform Codename} Platform Product Specification
- Reference design power on sequence block diagram (if included as part of the reference design schematics)
- Reference design power sequencing timings (if included in the platform design guide or a discrete document)
- Processor and chipset power sequencing timings

Embedded Controller Ecosystem

Hardware

Embedded Controller hardware is available from a number of vendors including but not restricted to:

- ITE Tech. Inc.*
- Nuvoton Technology Corp.
- Renesas Technology Corp.*
- SMSC*

Firmware

Development support for Embedded Controller firmware, including source code access, is available from the following 3rd party vendors.

Note: Please contact these vendors directly for a complete list of supported Embedded Controllers and firmware offerings.

Nuvoton Embedded Controller Firmware

- Nuvoton Technology Corp.: APC.Support@nuvoton.com

General Embedded Controller Firmware

- American Megatrends Inc.* (AMI): http://www.ami.com/products/product.cfm?ProdID=327&CatID=6&SubID=14
- Nanjing Byosoft Co., Ltd: http://byosoft.com.cn/en/EC.asp
- Phoenix Technologies Ltd.*: http://www.phoenix.com/pages/keyboard-controller-solutions

In addition to the firmware for the Embedded Controller, it is also necessary to have a system BIOS/EFI firmware that has the required hooks that facilitate interaction between the Intel Architecture aspects of the platform and the Embedded Controller. BIOS and EFI support is available from the following Independent BIOS Vendors (IBVs).

Independent BIOS Vendors*

- American Megatrends Inc. (AMI): http://www.ami.com/
- Nanjing Byosoft Co., Ltd: http://byosoft.com.cn/
- Insyde Software Corp.: http://www.insydesw.com/
- Phoenix Technologies Ltd.: http://www.phoenix.com/

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

Conclusion

This white paper has summarized what an Embedded Controller is on a conceptual level, the typical functions it typically performs, the specific devices that are found in Intel reference designs and the ecosystem in place to service these devices.

With this information, it is intended that a system designer has enough information to understand the role of an Embedded Controller in his or her system and how to proceed with its integration and customization of firmware as needed.

The Intel® Embedded Design Center provides qualified developers with web-based access to technical resources. Access Intel Confidential design materials, step-by step guidance, application reference solutions, training, Intel's tool loaner program, and connect with an e-help desk and the embedded community. Design Fast. Design Smart. Get started today. http://www.intel.com/p/en_US/embedded.

Authors

Matthew Lee is a Platform Application Engineer with the Low-Power Embedded Products Division (LEPD) at Intel Corporation.

Acronyms

AC	Alternating Current
ACPI	Advanced Configuration and Power Interface
ADC	Analog to Digital Convertor
BIOS	Basic Input/Output System
BOM	Bill of Materials
CPU	Central Processing Unit
CPLD	Complex Programmable Logic Device
DC	Direct Current
DDC	Display Data Channel
DVI	Digital Visual Interface
EC	Embedded Controller
EFI	Extensible Firmware Interface
FPGA	Field Programmable Gate Array

FWH	Firmware Hub
HDMI	High-Definition Multimedia Interface
I/O	Input/Output
IA	Intel Architecture
AMT	Intel® Active Management Technology
IBV	Independent BIOS Vendor
IC	Integrated Circuit
ICH	I/O Controller Hub
IP	Intellectual Property
KBC	Keyboard Controller
KSC	Keyboard and System Controller
LPC	Low Pin Count
ME	Management Engine
OS	Operating System
ODM	Original Design Manufacturer
OEM	Original Equipment Manufacturer
PCH	Platform Controller Hub
PECI	Platform Environment Control Interface
RAM	Random Access Memory
SMBus	System Management Bus
SMLink	System Management Link
SMC	System Management Controller
SPI	Serial Peripheral Interface
USB	Universal Serial Bus
VGA	Video Graphics Array

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. UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice.

This paper is for informational purposes only. THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Intel disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

§