

# PERSONAL COMPUTING DIVISION



## Phoenix ACPIArchitect™: Advanced Development Tool for Generating ASL Code

---

Phoenix Technologies combines extensive firmware knowledge with a leading position in the PC market to offer the most competitive and feature-rich ACPI BIOS products available today. Phoenix continues leading the industry in the transition to Microsoft's OSPM/ACPI model with the introduction of a powerful development tool, the **ACPIArchitect™**. The ACPIArchitect™ utility is an advanced development tool that is system BIOS independent and helps engineers in designing ACPI solutions for various platforms, including notebook, desktop, and server systems.

### **ACPIArchitect™ - Advanced Configuration and Power Interface Utility**

Phoenix has developed a powerful design tool called the ACPI Architect™. This tool includes chip set support, standard components, device and bus support, device and system templates, a built in code editor, a tree viewer, device icons, allows ease of differentiation and include on-line help.

### **ACPIArchitect™ - Utility to assist hardware and BIOS developers**

ACPIArchitect™ frees developers from learning the nuances of ASL and allows them to begin designing their systems immediately. ACPIArchitect™ is used to view, modify and create ACPI System Description Tables. The table view is presented to the user in the form of an ACPI Tree and the file is stored in ACPIArchitect™ form, \*.aa. The ACPIArchitect™ tool can generate ASL code from these \*.aa files. The ASL code can then be compiled using the Microsoft ASL compiler to generate AML code for use on ACPI systems.

The graphical user interface allows automatic code and template generation for devices, resources, and tables. Support for all ACPI-defined devices plus COM, LPT, DOCK, FDC, IDE, KBC, motherboard, audio, and other system components are supported. Table support includes FADT, APIC, SMB and Code SDT. System level support for embedded controllers and related library and on on-line help is included to make development simpler and allow designs to be ported more quickly.

### **User Interface that allow easy understanding and simplicity of use**

The ACPIArchitect™ operates in the Windows environment and behaves much like any other standard Windows application. The main components of the ACPIArchitect™ utility include:

**Menu Bar** - Provides access to all tools and functions available to ACPIArchitect.

**Tool Bar** - Provides quick access to standard windows file, edit print and help functions.

**Window(s)** - Each open ACPIArchitect file (\*.aa) is displayed in its own window as an ACPI Tree. Navigating the ACPI Tree is very like navigating in Windows Explorer. Click the + box to expand the tree beneath an item and the - box to collapse (hide) the tree

**Status Bar** - Displays the first line of ASL code for highlighted item.

## ACPI Tree

All ACPI hardware and features are organized as a tree structure. The tree represents how the operating system communicates with the devices. Objects under other objects in the tree are called children. The base of the tree is called the root. When the operating system begins to cut back power because it has detected an idle or non-use state, it does so from the bottom of the tree. This means that children are turned off before parent

The following is a portion of an ACPI tree created using ACPIArchitect™. In this tree, the IDE drive (IDE0) is a child of the PCI Bus (PCI0). In order to reduce power to the PCI Bus, the IDE drive (child) would have to be turned off before the PCI Bus (parent).



## ACPIArchitect™

ACPIArchitect™ is a Windows application written in Visual C++ 4.2. It provides an integrated Windows environment for developing ASL code. It is compliant with ACPI specification Revision 1.0. Using ACPIArchitect™ can greatly enhance productivity and reliability for ASL code development.

The main features of ACPIArchitect™ include:

1. Allows you to generate and edit a graphic object tree structure interactively, save it to file and load it from file.
2. Ability to generate the ASL code from the graphic tree structure and save it to a file. The generated ASL code can be compiled using Microsoft ASL compiler 1.0.
3. UnDo, Redo, Cut, Copy and Paste
4. Supports the following devices: COM port, Embedded Controller, Floppy, IDE, KBC, LPT port, Super I/O, USB and VGA. Also supports the following buses: PCI Bus, ISA Bus, ACPI Bus, EIO Bus; supports the following motherboard components: DMA controller, Interrupt Controller, Math-Coprocessor, Speaker, Timer, Real-Time Clock. Supports Fan, Lid, etc.
5. Allows you to customize configurations. Provides sample Method code so that minimal work is needed to generate ASL code for a specific platform.
6. Basic Library Support. The ACPIArchitect™ library will ultimately contain some predefined subsystems, much like the predefined devices (Insert - Devices) available in 1.0 only not limited to single discrete devices. At this time (version .7.1), users may create their own library items. This allows easy reuse of carefully designed systems.
7. Provides Help for key topics.