



Plug and Play BIOS Test Utility

[Phoenix Plug and Play BIOS Test Utility, Ver 1.07](#)

© Phoenix Technologies, Ltd., 1994,1999, All rights reserved

Настоящая утилита позволяет в интерактивном режиме выполнить функции PnP BIOS, задавая входные параметры функций и контролируя выходные параметры с помощью следующих функций

- | | |
|-------------------------------------|---------------------------------------|
| 0) Get # DNs | C) Get ISA Cfg_(40h) |
| 1) Get DevNode | D) Get ESCD Inf_(41h) |
| 2) Set DevNode | E) Read ESCD_(42h) |
| 3) Get Event | F) Write ESCD_(43h) |
| 4) Send Message | I) ISA PnP |
| 5) Get DockSta Info | L) Node List |
| 9) Set Static Res | S) Node Summary |
| A) Get Static Res | R) Real Mode |
| B) Get APM ID Table | P) Prot Mode |
| | Z) Header |

Function "0" - Get # DNs

Функция возвращает количество PnP Device Nodes и размер максимального блока данных PnP Node - Maximum Node Size.

PnP Device Nodes - это подмножество от множества PnP устройств, в которое входят устройства, известные BIOS, идентификаторы которых начинаются с сигнатуры "PNP". Список PnP Device Nodes готовится BIOS и, как правило, это список Onboard устройств, а не ISA PnP Cards, как ошибочно считают некоторые. Показательный пример - Onboard LPT Port включается в этот список, External LPT Port - не включается.

Предположительно, ситуация выглядит следующим образом: при проектировании платы заранее известно, что Onboard LPT удовлетворяет спецификации PNP устройства "LPT Port", (PNP0400 или PNP0401) поэтому в BIOS прописано, что этому порту необходимо присвоить статус PnP Node. Благодаря этому BIOS при вызове сервисных функций PnP Runtime Service может сообщить операционной системе о наличии порта, тип которого заранее известен, и операционная система не должна будет выполнять поиск и дополнительную идентификацию устройства. Относительно External LPT у BIOS информации меньше.

Если при сканировании пространства портов ввода/вывода было обнаружено, что, например, по адресу 3BCh читаемые данные совпали с записываемыми, то принимается решение о наличии порта LPT. Вносить его в список PnP Nodes нельзя, так как заранее неизвестно, что обнаруженное устройство соответствует спецификации PnP. Таким образом, главный критерий, по которому определяется принадлежность к System PnP Nodes - определение устройства в BIOS, особенности схемы устройства либо схемы подключения устройства к шине в расчет не принимаются.

Maximum Node Size - максимальный размер блока данных, который формирует функция Get PnP Node Info. Этот параметр нужен для системного программного обеспечения, вызывающего эту функцию и резервирующего определенный объем памяти для блока выходных параметров

[TOP](#)

Function "1" - Get DevNode

Функция возвращает информацию о заданном PNP Node. Необходимо ввести следующие параметры:

- Device Node Number - номер Device Node, нумерация с "0" включительно
- Control Code:
 1. Current = просмотр текущей конфигурации,
 2. Next Boot = просмотр конфигурации, которая должна быть установлена после перезагрузки

Подробнее о функции GetDevNode... (Under Construction!)

[TOP](#)

Function "2" - Set DevNode

Функция устанавливает и позволяет изменять информацию о заданном PNP Node. Необходимо ввести следующие параметры:

- Device Node Number - номер Device Node, нумерация с "0" включительно
- Control Code:
 1. Current = просмотр текущей конфигурации,
 2. Next Boot = просмотр конфигурации, которая должна быть установлена после перезагрузки
 3. Both = оба вида конфигурации

После ввода параметров, последовательно выводятся значения байтов из конфигурационных полей, формат которых подробно описан выше, в описании функции Get DevNode, и пользователь может ввести новые параметры.

Экспериментальные данные:

С помощью этой функции удалось переопределить порт COM1=3F8h на адрес COM2=2F8h для текущей конфигурации (COM2 при этом был запрещен в CMOS Setup). Переопределение состоялось не только в смысле изменения состояния переменных BIOS. Сканирование пространства портов ввода/вывода показало, что по адресу 3F8h регистровый файл исчез, а по адресу 2F8h появился. Следовательно, PnP BIOS перепрограммировал конфигурацию MIO Chip, открывая возможность с помощью вызовов PnP Runtime Service выполнять аппаратно-зависимые операции единым для всех систем методом.

Очевидно, что с External COM на ISA карте BIOS не может делать таких операций. Это согласуется с высказанным в начале документа предположением о способе назначения статуса System PnP Node.

Выполнить конфигурирование для Next Boot не удалось, видимо из-за того, что для этого при перезагрузке пришлось бы "ослушаться" установок Setup для COM порта

[TOP](#)

Function "3" - Get Event

На нескольких испытываемых платах возвращалась ошибка 82h - Function Not Supported. Предположительно, прототипом для этой функции является функция 3 PnP Runtime Service (Get Event), которая выдает системному программному обеспечению информацию о событиях, связанных с подключением и отключением устройств без отключения питания и перезапуска, например USB.

Требуются дополнительные испытания с использованием различных платформ

[TOP](#)

Function "4" - Send Message

На нескольких испытуемых платах возвращалась ошибка 82h - Function Not Supported. Предположительно, прототипом для этой функции является функция 4 PnP Runtime Service (Send Message), которая позволяет системному программному обеспечению сообщить PnP BIOS о событиях, связанных с подключением и отключением устройств без отключения питания и перезапуска

[TOP](#)

Function "5" - Get DockSta Info

На нескольких испытуемых платах возвращалась ошибка 82h - Function Not Supported. Предположительно, прототипом для этой функции является функция 5 PnP Runtime Service (Get Docking Station Information). Коды функций 6,7,8 пропущены, так как эти номера функций зарезервированы в PnP BIOS.

[TOP](#)

Function "9" - Set Static Res

Предположительно, имеет место ошибка в документации - эта функция выводит список ресурсов, поэтому должна называться Get Static Res

Теоретически, Static Resources - это список системных ресурсов, используемых ISA картами не соответствующими спецификации PnP (Legacy ISA Cards). Под системными ресурсами подразумеваются линии IRQ, DRQ, адреса памяти и пространство адресов ввода/вывода. Значения Static Resources не могут программно изменяться. PnP BIOS должен иметь информацию о Static Resources, для того, чтобы исключить эти ресурсы из списка динамически распределяемых, например, чтобы исключить назначение PnP устройству линии IRQ, которую использует Legacy устройство.

Экспериментальные данные:

Вывод списка сопровождается сообщением об ошибке взаимодействия с ESCD. Изменения содержимого списка при изменении конфигурации Legacy Cards не происходит. Так как PnP BIOS "не знает" Non-PnP Legacy устройства, он не может автоматически определить, какие ресурсы они используют. Видимо для того, чтобы информация в списке была корректной, системное программное обеспечение должно выполнить операцию Set Static Resources. Это может сделать операционная система, получив список ресурсов, используемых Legacy устройством от драйвера этого устройства, либо пользователь, в режиме интерактивного редактирования ESCD

[TOP](#)

Function "A" - Get Static Res

Если предположение об ошибке верно, то эта функция - Set Static Resources. Операция Set Static Resources (прототип которой функция 9 PnP BIOS Runtime Service) предназначена для того, чтобы сообщить PnP BIOS, какие ресурсы используются Non-PnP Legacy картами (см. описание [Function "9"](#)). В PnP BIOS эта функция использует не унифицированный формат передачи параметров, который может быть различен у Award, AMI, Phoenix. Возможно, по этой причине при запуске происходит ошибка

[TOP](#)

Function "B" - Get APM ID Table

Прототипом этой функции является функция 0Bh (Get APM ID Table) PnP BIOS, которая возвращает таблицу формата:

```
DWORD PnP Device Identifier #1
WORD APM Identifier #1
...
DWORD PnP Device Identifier #N
WORD APM Identifier #N
```

Настоящая функция позволяет системному программному обеспечению установить соответствие между PnP устройствами и APM (Advanced Power Management) идентификаторами

[TOP](#)

Function "C" - Get ISA Cfg (40h)

Прототипом этой функции является функция 40h PnP BIOS (Get PnP ISA Configuration Structure). Возвращаются следующие параметры:

ISA Config Revision	Номер версии структуры (Structure Revision)
Number of CSNs assigned	Количество присвоенных CSN (Card Select Number), количество обнаруженных ISA PnP карт
ISA Read Data Port	Параметр достоверен только тогда, когда в системе имеется хотя бы одна ISA PnP карта

Для доступа к ISA PnP карте используется 3 порта:

ADDRESS PORT	Порт для записи адреса регистра. Адрес совпадает с LPT2 Status. Конфликта не происходит, так как LPT Status - Read Only, а PNP Address Port - Write Only	0279h
WRITE DATA PORT	Порт для записи данных в PnP регистр. Адрес алиасный к LPT Data 0279h	0A79h
READ DATA PORT	Порт для чтения данных из PnP регистра, ему динамически назначается адрес в диапазоне 0203h-03FFh. При выполнении POST выбирается адрес, не конфликтующий с другими устройствами. Адрес этого порта выводится как ISA Data Read Port	0203h-03FFh

[TOP](#)

Function "D" - Get ESCD Inf (41h)

Прототипом этой функции является функция 41h PnP BIOS (Get ESCD Info), которая возвращает информацию о блоке параметров ESCD (Extended System Configuration Data), хранящегося в энергонезависимой памяти NVRAM. Область хранения - это обычно специальная область в BIOS Flash ROM; в EISA системах это может быть статическое CMOS RAM с батарейным питанием.

Возвращаются следующие параметры:

```
NVRAM Size - размер блока памяти NVRAM
ESCD Size - размер блока параметров ESCD
NVRAM Base - базовый 32-разрядный адрес блока памяти NVRAM
```

Если NVRAM не включено в пространство памяти или недоступно через порты с индексным доступом, параметр NVRAM Base = 0

[TOP](#)

Function "E" - Read ESCD (42h)

Прототипом этой функции является функция 42h PnP BIOS (Read ESCD). Функция позволяет прочитать данные, находящиеся в ESCD и выдает их в виде дампа. Экспериментально установлено, что выполнение достоверно, только если перед этим вызывалась [функция "D"](#)

[TOP](#)

Function "F" - Write ESCD (43h)

Прототипом этой функции является функция 43h PnP BIOS (Write ESCD).
Позволяет записывать данные в блок ESCD, перед вызовом этой функции должны быть отработаны [функция "D"](#) и [функция "E"](#).
Не удалось разобраться, как ввести записываемые параметры или отредактировать имеющийся дамп

[TOP](#)

Function "I" - ISA PnP

Функция имеет три подфункции:

- Display CSN and Read Data Port
Отображает следующие параметры: Maximum CSN - количество ISA карт (Card Select Numbers); Read Data Port - адрес порта PnP Read Data (см. пояснение в [Function "C"](#)). Также выводится список PnP устройств с идентификаторами (Алгоритм получения 7-символьного ASCII идентификатора из 32-битового упакованного идентификатора приведен в описании [Function "1"](#))
- Re-Isolate Cards
Выполняется процедура, в ходе которой определяется бесконфликтное значение адреса PnP Read Data Port и каждой PnP карте присваивается CSN (Card Select Number)
- Send Init Key and use POST CSN
Send Init Key - процедура передачи специальной байтовой последовательности (ключа), при получении которой PnP устройства входят в режим конфигурирования.
Use POST CSN - использование CSN, присвоенных при выполнении POST для дальнейшего взаимодействия с PnP устройствами

[TOP](#)

Function "L" - Node List

Выводит список PnP Nodes, сформированный BIOS.

Size	Размер информационного блока PnP Node в байтах
Num	Номер PnP Node в списке, составленном BIOS
ID	Идентификатор, распакованный из 32-битового Compressed ID
Type Codes	Байты Base Class / Sub Class / Programm Interface Level
Attribytes	Атрибуты устройства
ANSI ID	Текстовая строка, идентифицирующая устройство (опционально)

Пояснения по всем параметрам приведены в описании [Function "1"](#)

[TOP](#)

Function "S" - Node Summary

Выводит список PnP Nodes и информацию об используемых системных ресурсах для каждой из них

IRQ _i	означает использование линии запроса прерывания с номером <i>i</i>
DRQ _i	означает использование линии запроса DMA с номером <i>i</i>
(X1-X2)#Y	означает диапазон в пространстве ввода-вывода с минимальным программно-конфигурируемым адресом X1, максимальным X2 и размером диапазона Y
Mem X#Y	означает диапазон в пространстве памяти с адресом X и размером Y

[TOP](#)

Function "R" - Real Mode

Задаёт использование Real Mode интерфейса для вызова функций PnP BIOS при выполнении последующих операций

[TOP](#)

Function "P" - Protected Mode

Задаёт использование Protected Mode интерфейса для вызова функций PnP BIOS при выполнении последующих операций. Функции PnP BIOS доступны в двух интерфейсах:

Real Mode - реальный режим
Protected Mode - 16-битовый защищенный режим

Подразумевается, что 32-битное системное программное обеспечение будет использовать 16-битные вызовы для функций PnP BIOS. При межсегментном вызове процедуры, целевой сегмент должен иметь Default Operand Size = 16, Default Address Size = 16, параметры, передаваемые через стек, должны быть 16-битовые

[TOP](#)

Function "Z" - Header

Выводится содержимое блока PnP BIOS Installation Structure, который находится в области BIOS Runtime Code F000:0000h...F000:FFFFh по базовому адресу, кратному 16 и имеет длину 33 байта. Блок содержит параметры, которые использует системное программное обеспечение для взаимодействия с PnP BIOS. Наличие такого блока в указанном диапазоне адресов означает наличие PnP BIOS, что используется при определении.

Signature	сигнатура, по которой опознается управляющий блок PnP, "\$PnP"
Revision	номер версии, упакованное BCD число, 10h означает Version 1.0
Length	длина структуры в байтах
Checksum	Control Field - управляющее поле контрольная сумма
Event Flag Address (Phys)	физический 32-разрядный адрес Event флага
Real Mode Entry	адрес точки входа для Real Mode Runtime Service
Protected Mode Entry	адрес точки входа для Prot Mode Runtime Service
OEM ID	идентификатор OEM
Real Mode Data Seg	сегмент данных для Real Mode Runtime Service
Prot Mode Data Seg	сегмент данных для Protected Mode Runtime Service

Формат PnP BIOS Installation Structure:

00h	BYTES	Сигнатура "\$PnP" = 24h, 50h, 6Eh, 50h
04h	BYTE	Номер версии, BCD. 10h означает Version 1.0
05h	BYTE	Длина блока, равна 21h=33
06h	WORD	Управляющее поле, задает тип Event Notification 00=Not Sup 01=Use Polling 10=Asynchronous
08h	BYTE	Контрольная сумма. Значение этого байта должно быть таким, чтобы сумма по модулю 256 для блока, длина которого указана в байта по смещению 5, была равна нулю. В противном случае структура не считается достоверной
09h	DWORD	Адрес флага Event Notification (физический 32-разрядный)
0Dh	WORD	Смещение Real Mode Entry Point
0Fh	WORD	Кодовый сегмент Real Mode Entry Point
11h	WORD	Смещение Protected Mode Entry Point
13h	DWORD	Базовый адрес 16-битового кодового сегмента Protected Mode Entry Point
17h	DWORD	Идентификатор OEM Device ID
1Bh	WORD	Сегмент данных для Real Mode
1Dh	DWORD	Базовый адрес сегмента данных для Protected Mode

Для FAR вызова подпрограммы PnP BIOS Runtime Service Procedure используются сегмент и смещение. Представленные в таблице сегментные адреса для Real Mode используются непосредственно как значения сегментных регистров. Сегментные адреса для Protected Mode используются как значения SegmentBase в дескрипторах сегмента, а в сегментный регистр загружается селектор, который ссылается на данный дескриптор. Значение селектора вызывающая процедура выбирает по собственному усмотрению

[TOP](#)