Designing Multiple Function PC Cards

National Semiconductor
Application Note 975
January 1995

- Sharing the IREQ Signal
- Mapping I/O Port Ranges
- Configuration Registers
- Multi-function PC Card CIS
- Interrupt Sharing Protocol
- Arbitration
- Power Management
- Common Memory
- LAN Operation

OVERVIEW

The broad acceptance of portable computers has driven a need to add functionality to these systems in a portable manner. PCMCIA has addressed this need by specifying a mechanism for placing a given I/O application on a PC Card. Since there are typically only two slots in a computer, however, there's a growing need to place two or more I/O applications on the same card.

This application note discusses the challenges implementing multiple I/O function PC Cards. Modifications or extensions to the PCMCIA PC Card Standards are discussed. All of the changes are intended to have minimal impact on host system software or socket hardware. The bulk of the changes concern the PC Card interface and Card Information Structure. Changes to the Card Services interface are also proposed. No changes are required for the Socket Services interface or socket controllers. The intent is to maximize backward compatibility with existing and future host systems.

WHY PC CARD STANDARDS NEED MODIFICATION

Multiple function PC Cards may be implemented with the current PC Card standard. However, there is no definition how such cards should be implemented. This forces any client device drivers that wish to use the functionality on the PC Card to contain specific knowledge of the card. It is not possible to implement a generic multiple function PC Card client device driver given the wide variation in implementations that could be used.

Requiring any use of multiple function PC Cards to require a specific client device driver severely limits the interoperability of such a PC Card. Before it could be used on a host platform, the specific client device driver would have to be created for the combination of the card and the host environment. Instead of allowing the host platform manufacturer to develop a generic handler for all cards providing the card's functionality, another client device driver would be required. If an end-user wished to use the multiple function card and a single function card delivering some of the same capabilities, both drivers would have to be available in the host system, potentially wasting available system resources.

This proposal does not require that all multiple function PC Cards provide their functionality as described by this document. A card manufacturer that is willing to be totally responsible for all the client device drivers that use their card could choose to implement in any manner they chose. This proposal is intended to define a standard means of implementing a multiple function PC Card to allow generic client device drivers to be developed promoting the interoperability of such a card.

Before exploring the implementation of multiple function PC Cards, this document reviews the handling of single function cards and cards which combine memory and a single I/O function.

SINGLE FUNCTION PC CARDS

In most cases today, PC Cards provide a single expansion function for a host system. For example, linear memory storage, data/FAX modem, LAN adapter or ATA disk drive. When a PC Card is inserted in a socket, software on the host system dedicated to the single function (often provided with the card) or a generic enabler recognizes and configures the card. Further, the present Card Services interface specification limits PC Card configuration to a single client. The specification does allow multiple memory clients to share a partitioned memory card, but the use of memory within a partition is expected to be exclusive.

MEMORY AND I/O PC CARDS

Combining memory storage devices with a single I/O function is well-defined and easily achieved within the definition provided by the existing PCMCIA PC Card Standards. However, even this limited sharing can be complicated if separate memory and I/O clients require the ability to control voltages delivered to the card. The current Card Services interface only allows a single client to control voltages applied to a PC Card.

MULTIPLE I/O FUNCTION PC CARDS

Combining multiple I/O functions on a single PC Card is more problematic than the uses described above. There are a number of issues. They include:

- Describing multiple functions in the PC Card's Card Information Structure.
- Allowing single function host software to utilize one function on a multiple function PC Card without affecting other functions.
- Allowing multiple I/O functions to share the single PC Card IREQ signal.
- Mapping individual I/O port ranges used by each function into host system address space.
- Control of V_{CC} and V_{PP} settings.

The following sections expand on these issues and pose a number of questions.

PROCOMM PLUS® is a registered trademark of DATASTORM, TECHNOLOGIES, INC.

MULTIPLE FUNCTION CARD INFORMATION STRUCTURES

The present PCMCIA PC Card Standard does not describe how the Card Information Structure (CIS) of a multiple function PC Card should be formatted. The current specification assumes a single set of configuration registers with a single configuration index. There is no way to specify in the CIS which configuration index values apply to the functions on the PC Card.

SEPARATE FUNCTION ENABLING

The use of a single set of configuration registers on a PC Card introduces a number of side effects when multiple clients attempt to use a card. For example, when a modem client enables modem functionality, how does a network adapter client later enable the network functionality?

In addition, which client gets to report status through the Pin Replacement and Configuration and Status Registers? If they both have interrupts pending, which is reporting?

SHARING THE IREQ SIGNAL

When two or more functions use the same IREQ signal, the host system needs to determine which function is interrupting and be able to detect when other functions also require servicing. There are a number of methods that might be employed and this can be less of a problem on systems capable of level interrupt triggering. However, PC Cards are often used in systems using edge-triggered interrupts and sharing in those environments must also be supported. With one of the design goals requiring no new hardware in the host system, the proposed solution must be limited to changes in PC Card hardware.

Since there is a single IREQ line for a PC Card, which function is reporting when an interrupt event is signaled? Which function handler receives the interrupt event notification in the host system? Which function handler performs required End Of Interrupt processing on the host system?

MAPPING I/O PORT RANGES

If a PC Card contains multiple functions, where are the I/O registers for each function mapped into host system space? Some x86 systems have conventions for the location of function specific registers (i.e. serial ports used for data/FAX modems) if these cards are to be used with existing software (i.e., PROCOMM PLUS®). The registers for other functions on the card are usually not located in contiguous address space due to the fact that such space may be in-use by another PC Card or host-based peripheral. How will non-contiguous address ranges be handled and how many need to be handled?

CONTROL OF V_{CC} AND V_{PP}

This problem is fairly straight forward, the current standard only allows a single client to control voltages on a PC Card. This may not be sufficient if multiple clients are using the card and each requires such control.

DESIGN GOALS

As noted above, multiple function PC Cards will enjoy wider acceptance if specific card enablers and drivers are not required. Based on the issues listed above, there are several design goals for supporting multiple function PC Cards. They include:

- Tie function specific information to configuration information in the Card Information Structure.
- Allow independent use of multiple functions on the card by multiple clients including the ability to individually enable, control, disable and report status for each function.
- Since there is a single IREQ line, develop an interrupt sharing protocol that allows multiple functions on the PC Card to share interrupt notifications. Support existing software that may be unaware of interrupt sharing.
- Avoid requiring protocol for inter-client communication in order to share a PC Card outside of the current programming model used by Card Services.
- Do not require any change to socket hardware as currently built.
- Minimize impact to existing system software. If existing software may not be used with a multiple function PC Card, do not cause existing software to malfunction if it encounters such a card.

GENERIC MULTIPLE I/O FUNCTION PC CARDS

This section is an overview of the proposed implementation for multiple I/O function PC Cards.

SEPARATE CONFIGURATION REGISTERS FOR EACH FUNCTION

To eliminate interaction between multiple functions of a PC Card, each function has its own set of configuration registers. In this manner, host software interacts only with the registers related to the function being managed and each function may be completely independent.

Each function specific Configuration Option Register uses individual bits to signal function specific configuration information:

Bit 0–LSB	Function specific enable.
Bit 1	Enable function specific Base and Limit Registers
Bit 2	Enable function specific IREQ routing
Bit 3–5	Reserved for vendor implementation

Each function specific Configuration and Status Register defines Bit 0 as **IntrReset.** When this bit is reset to zero (0), the function handles interrupt notification the same as existing implementations. The **Intr** flag (Bit 1) represents a function specific interrupt request and is cleared by the function on the card when the event has been serviced.

When **IntrReset** is set to one (1), functions require a positive acknowledgment when the host system has completed processing the interrupt event and the system is ready to accept another interrupt notification. The system notifies the PC Card it is ready for another interrupt by resetting the **Intr** flag to zero (0). If the function (or any function on the PC Card) is signaling another interrupt event, the card must again notify the host system of an interrupt condition. Since the card must signal if any function on the card that has interrupts enabled requires service, the **Intr** bit is aliased to all functions having the **IntrReset** bit reset to zero (0).

ADDITIONAL CONFIGURATION REGISTERS

Additional registers are defined following the I/O Event Register. These include up to four Base Address Registers and a single Limit Register. These registers are written by host software during function configuration. The base of the range of contiguous I/O registers used by the function in host system address space is written to the Base Address registers before the function is enabled by writing the Configuration Index. This allows host software to place the function wherever it desires in host system space. The Limit

Register describes the size of the I/O range used by the function. Each bit in this register represents an address line, and all bits to the right of any bit set to one (1) must also be set to one (1). This requires I/O ranges to be sized as a power of two.

MULTI-FUNCTION PC CARD CIS

Multiple function PC Cards have multiple Card Information Structures (CIS). The first or primary CIS on a PC Card identifies the card as containing multiple functions by the presence of a CISTPL—LONGLINK—MFC tuple and the absence of the CISTPL—FUNCID tuple. The PC Card also has a separate secondary CIS for each set of Configuration Registers on the card.

The starting location of each secondary CIS is described in a single CISTPL—LONGLINK—MFC tuple in the first CIS. Each additional CIS shall begin with a CISTPL—LINKTARGET tuple. Each secondary CIS will contain a CISTPL—FUNCID tuple.

The presence of the Base Address and Limit Registers are noted in the Configuration Tuple Configuration Registers Presence Mask within each secondary CIS. An implementation may choose to limit the number of Base Address Registers by resetting corresponding bits to zero.

For most environments, at least two Base Address Registers are required to allow I/O ranges to be located within a 64 KByte I/O address spaces. The single Limit Register may be eliminated if all possible configurations for a function use the same number of I/O address lines.

INTERRUPT SHARING PROTOCOL

Each function supports enabling and disabling interrupt notification through its Configuration Register using Bit 2. In addition, each function uses the **Intr** and **IntrReset** bits as described above.

When multiple functions on a PC Card use interrupt notifications on the card's single IREQ line, it is important that End Of Interrupt (EOI) processing be performed only once. The preferred method of insuring that EOI processing happens only once is to have Card Services receive all card interrupt notifications and dispatch them for function specific processing to registered handlers. When all handlers have completed processing, the Card Services handler performs EOI processing, resets the **Intr** line on the card and returns from the interrupt notification.

Interrupt handlers for multiple function PC Cards are registered with Card Services using an extended **RequestIRQ** function. The argument packet contains one additional four-byte field for specifying the callback handler for the client's interrupt handler. If the request is successful, Card Services sets the **AssignedIRQ** field in the **RequestIRQ** packet to the assigned IRQ level, which may already be in-use by other functions on the PC Card. Interrupts from the PC Card transfer control to the Card Services interrupt handler. The Card Services handler then transfers control to the entry point registered by the client using the **RequestIRQ** function via a CALL instruction. The client handler processes the notification and returns control to the Card Services handler by executing a RET instruction.

While processing the interrupt notification, the client interrupt handler may enable interrupts on the host system. The client interrupt handler may also modify any other host system registers. The Card Services interrupt handler restores all registers to their entry value before returning from the interrupt notification.

Interrupt handling is more complicated if the enabling client is only requesting interrupt routing on behalf of another software entity on the host system that may not have been loaded. For example, a generic enabler might configure the data/FAX modem function on a PC Card to use a standard COM port definition expecting a communications program loaded sometime later to simply use the port without realizing the port resides on a PC Card.

In this case, Card Services must utilize a redirection routine that transfers control to a traditional interrupt handler by simulating the hardware event. Since the traditional interrupt handler performs its own EOI processing, the Card Services handler must not perform EOI processing. A new flag in the **Attribute** field indicates the interrupt routine is performing EOI processing. Only one client may set this flag to one (1).

The redirection routine is responsible for determining if it is appropriate to transfer control to the simulated interrupt handler. It is possible that no handler has yet been registered for the simulated interrupt event. Typically, the redirection routine checks the Interrupt Mask Register (IMR) on the Programmable Interrupt Controller (PIC) to confirm the simulated level has been unmasked.

WARNING: Simulated interrupt handlers performing End Of Interrupt (EOI) processing MUST use non-specific EOI notifications to the Programmable Interrupt Controller.

NATIONAL PCM16C00VNG IC

To simplify the development of Multi-Function PC Cards, National Semiconductor Corporation (National) has developed a Standard Integrated Circuit known as the PCM16C00VNG. The PCM16C00VNG provides the necessary silicon to interface two (2) independent I/O functions on a PC Card to a PC Card socket. It also provides access to memory devices on the PC Card including non-volatile storage of a Card Information Structure (CIS) in inexpensive serial EEPROM devices. The following is a list of the PCM16C00VNG's key features.

- Support for two (2) completely independent I/O functions allowing separate enabling with generic client device drivers.
- Support for memory devices in either the attribute or common memory planes.
- Full compliance with the PCMCIA PC Card Standard and proposed Multi-Function Extension minimizing system software impacts. This includes the IREQ generation protocol outlined in the proposed Multi-Function Extension.
- Full functionality with existing socket hardware designs, making current systems more capable with a simple PC Card system software update.
- CIS storage in inexpensive serial EEPR0Ms, automatically transferred to write-protected RAM within the PCM16C00VNG during initialization.

- Automated EEPROM-based CIS update through simple RAM accesses, easy and reliable field updates by end-users.
- Programmable digital port provides function-specific, in-the-field customization.
- Independent, programmable power management for each function allows automatic or on-demand power conservation.
- Card side DMA transfer support for devices such as the National DP83902VJG Ethernet Controller for supporting shared memory and programmed I/O mode drivers.
- Programmable arbitration unit allows bus-mastering, on-card functions to be performance-tuned after manufacture by simply updating the CIS.
- On-board buffer and glue logic reduces I/O and memory function interface requirements to maximize the use of PC Card real estate.

The PCM16C00VNG manages a common data bus and common address bus on the Multi-Function PC (MFPC) Card. The chip also provides control signals for both I/O functions and access to common memory. These control signals include function access, event signals, reset, power management, and arbitration if bus mastering is required for common memory access. *Figure 1* is a block diagram of the PCM16C00VNG used in a PC Card design.

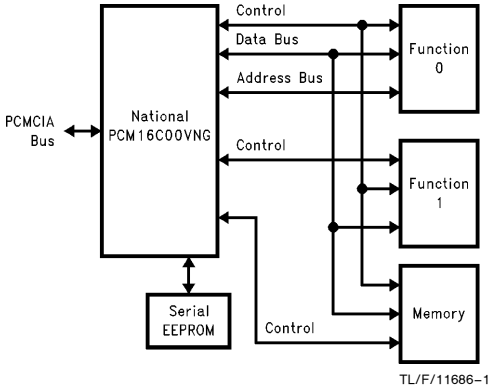


FIGURE 1. National PCM16C00VNG Connection Block Diagram

I/O FUNCTION SIGNALING

Between the PCM16C00VNG and each I/O function are signals for chip selection, interrupts, wait or bus throttling, ready status, reset, bus width (8- or 16-bit), read/write, and external asynchronous events. As with peripherals attached to an x86 Industry Standard Architecture (ISA) bus, data width is determined by the size of the access and the function's use of the IOCS16 signal.

Each I/O function signals interrupt events as per the proposed Multi-Function extension to the PC Card Standard. If only one device is using interrupts, the PCM16C00VNG may be programmed to create interrupt events as currently produced by single-function PC Cards. The PCM16C00VNG also supports the positive-acknowledgment protocol defined for sharing the single IREQ signal from the PC Card.

ARBITRATION

Some I/O functions benefit from bus mastering or DMA transfers. The PC Card Standard, Revision 2.10, does not specify these types of transfers across the interface between a PC Card and a host system. However, the PCM16C00VNG allows I/O functions to perform such transfers to memory devices located on the PC Card. The host system may then make potentially more efficient memory to memory transfers.

Queuing parameters and requirements for such transfers vary widely among I/O functions. To accommodate these differences, the PCM16C00VNG allows the arbitration characteristics such as priority, latency, and preemption to be programmed. Since most designs benefit from actual experience, the PCM16C00VNG allows these parameters to be adjusted at any time in a product's life cycle by simply updating the CIS.

POWER MANAGEMENT

As with other control capabilities, the PCM16C00VNG provides independent management of power conservation for each I/O function. If an I/O function can recover transparently from power down, the PCM16C00VNG may be programmed to place a function in power-down mode during periods of inactivity. When the PCM16C00VNG senses a function-specific access, it automatically releases the function from its power-down state and allows the host access to complete.

Host-based software may select other power conservation states by requesting that the PCM16C00VNG stop or divide clock signals for a specific I/O function. The PwrDwn bit of the PCMCIA-defined Card Configuration and Status Register is also supported.

COMMON MEMORY

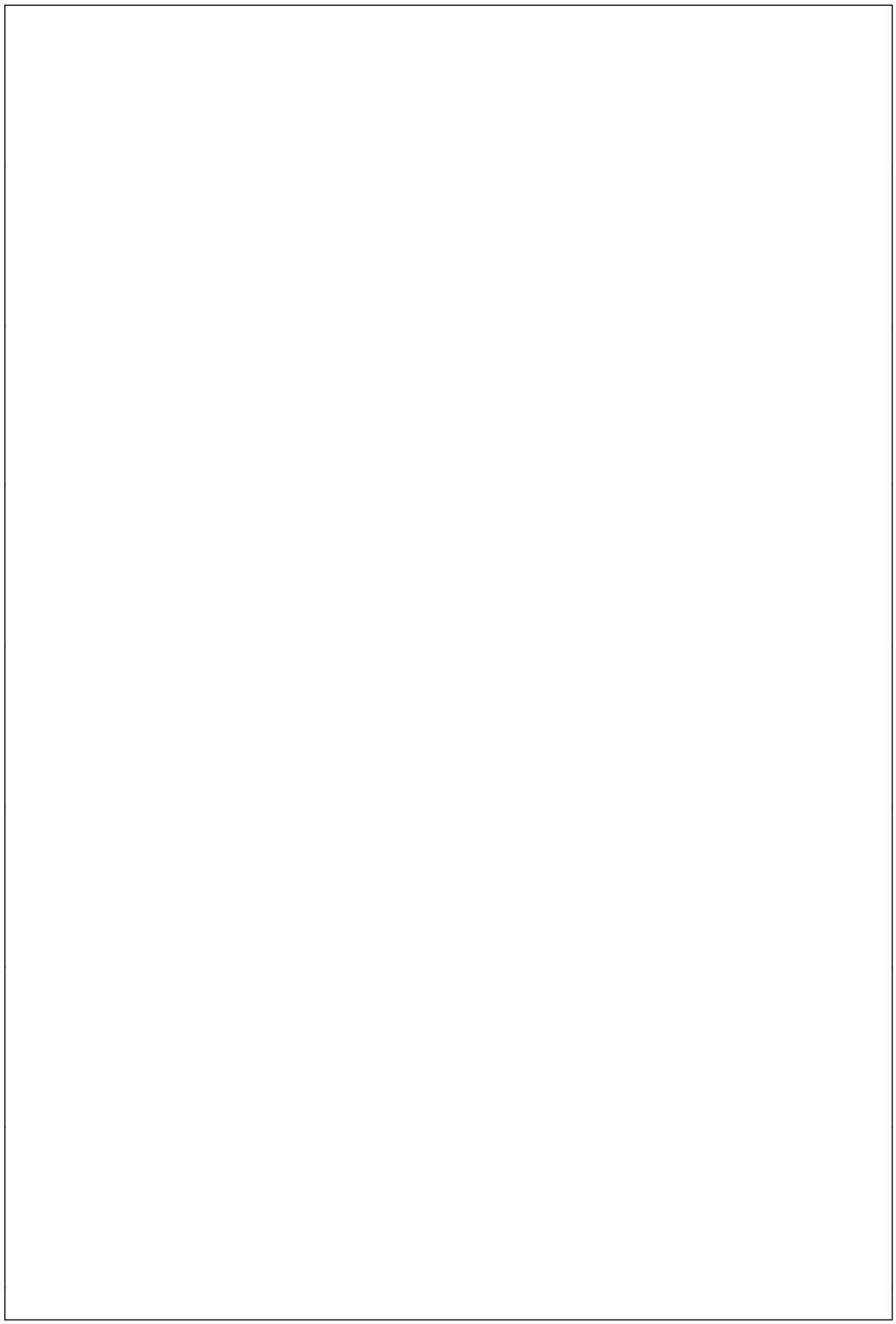
Connecting common memory devices is straightforward. The PCM16C00VNG passes common memory accesses without decoding. Additional address lines beyond those supported by the PCM16C00VNG are simply routed directly to the memory devices. The PCM16C00VNG confirms that any card-based bus mastering devices are inactive before allowing a host access to proceed. Memory functions requiring split address ranges can fragment common memory into two discontiguous ranges using a single address line without glue logic.

LAN OPERATION

The PCM16C00VNG incorporates special circuitry to support National's DP83902VJG Ethernet controller without any glue logic. This support allows both shared DMA (Shared Memory Mode Software Drivers) and remote DMA (Programmed I/O Mode Software Drivers) operation. The arbitration unit within the PCM16C00VNG makes these DMA modes possible. In addition, the PCM16C00VNG provides additional I/O registers within the device when programmed for LAN Mode operations. These registers provides support for coaxial or twisted-pair Ethernet applications and other functions.

ACKNOWLEDGEMENTS

National would like to thank Dave Lawrence from Ventura Micro Inc. for his contribution to this application note.



LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: 1(800) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-Gargan-Str. 10
D-82256 Fürstenfeldbruck
Germany
Tel: (81-41) 35-0
Telex: 527649
Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
Sumitomo Chemical
Engineering Center
Bldg. 7F
1-7-1, Nakase, Mihama-Ku
Chiba-City,
Ciba Prefecture 261
Tel: (043) 299-2300
Fax: (043) 299-2500

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductores Do Brazil Ltda.
Rue Deputado Lacorda Franco
120-3A
Sao Paulo-SP
Brazil 05418-000
Tel: (55-11) 212-5066
Telex: 391-1131931 NSBR BR
Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
Building 16
Business Park Drive
Monash Business Park
Nottinghill, Melbourne
Victoria 3168 Australia
Tel: (3) 558-9999
Fax: (3) 558-9998