V-ACPI Functional Specification

1. Revisions

Time/Date	By	Comments
1/71997	John Kabat	• Changed definitions of **setup_idx** and **setup_data** for GPIOx assignment • Defined a setup_idx (0x41) to allow assignment of the SCI IRQ
12/23/1997	Jim Kelsey	• Added Terminology Section • Added Chapter 6 (V-ACPI Theory Of Operation) • Moved SMM specifics to Chapter 6.
12/18/97 2:34 PM	John Kabat	• Restored P_LVL3 register. • Added SETUP_IDX and SETUP_DATA registers. • Added GPIO to _STS bit mapping.
10/13/97 1:05 PM	John Kabat	• Removed P_LVL3 register. • Reordered registers for future support by hardware in Kaluha • Changed bit assignments in P_CNT to enhance hardware support in Kaluha
9/25/97 3:00 PM	John Kabat	• Removed PM2_CNT register, as we only will support single processor systems. • Realigned registers to conserve address space. • Minor editing changes.
9/9/97 9:30 AM	John Kabat	• Revised schedule to include testing. • Modified P_CNT register to show values for CPU duty cycle. • Added internal test registers for test programs.
9/5/97 11:10 AM	John Kabat	• Revised to reflect ONLY SMM items.
9/3/97 4:08 PM	John Kabat	• Changed Schedule to add dates and identify Phoenix BIOS as 1st BIOS to implement
9/2/97 3:34 PM	John Kabat	• Fixed Definition of bits resets in PM1A_STS register
8/22/97 2:00 PM	John Kabat	• Corrected ACPI Fixed Feature Space Register Layout to include proper block grouping. • Added time for BIOS Study. • Added indication that certain tasks are BIOS dependent and must be performed for each BIOS. • Corrected error in BM_STATUS bit definition in PM1a_STS register

2. Terminology

ACPI	Advanced Control Power Interface
Cx	CPU Power State. Possible CPU power states are C0-C3.
Dx	Device Power State. Possible device power states are D0-D3.
GPIO	General Purpose Input/Output
SMI	System Management Interrupt
SMM	System Management Mode
SM-RAM	System Management RAM
Sx	System Power State. Possible system power states are S0-S4.
V-ACPI	Virtual ACPI
VSA	Virtual System Architecture
VSM	Virtual Support Module

3. Introduction

ACPI is an emerging specification that defines an interface that enables the ACPI-aware operating system to directly and efficiently manage the configuration and power consumption of the system's CPU and devices. An Operating System Power Management (OSPM) driver enforces power management *policy* by communicating directly with the ACPI chipset and an optional embedded controller.

An ACPI compliant system is one whose underlying BIOS, device drivers, chipset and peripherals conform to revision 1.0 or newer of the Advanced Control & Power Interface specification. This document outlines Cyrix' solution for implementing the chipset and BIOS components of an ACPI compliant system.

In an ACPI compliant Cyrix MediaGx platform, Fixed Feature and General Purpose registers are virtual -- these registers are emulated by the SMI handling code rather than existing in physical hardware. To the ACPI compliant operating system, the SMI-base virtualization is transparent.

For more details regarding the V-ACPI theory of operation, consult Chapter 6.

Cyrix' Cx5520/5530 V-ACPI solution provides the following support:

CPU States	C1, C2
Sleep States	S1, S2, S4, S4BIOS, S5
Embedded Controller (Optional)	SCI and SWI event inputs.
General Purpose Events	Fully programmable GPE0 Event Block registers.

4. V-ACPI Fixed Feature Space Registers

Fixed Feature space registers are required to be implemented by all ACPI-compatible hardware. The Fixed Feature registers in the VSA/ACPI solution are mapped to normal I/O space starting at offset AC00H. Each register within this I/O space resides on a four-byte boundary and must be accessed as such. Unaligned accesses to Fixed Feature registers are **not** supported.

```
ACPI_FIXED_REG_BASE. – 0xAC00
```

All offsets are relative to ACPI_FIXED_REG_BASE except where noted.

Offset	Align	Len	Name	Location/Description
0:3	4	4	P_CNT	P_BLK
4	1	1	P_LVL2	P_BLK
5	1	1	P_LVL3	P_BLK
6	1	1	SMI_CMD	ACPI Enable/Disable Port
7	1	1	Reserved	
8:9	2	2	PM1a_STS	PM1a_EVT_BLK
10:11	2	2	PM1a_EN	PM1a_EVT_BLK
12:13	4	2	PM1a_CNT	PM1a_CNT_BLK
14:15	2	2	SETUP_IDX	V-ACPI internal index register.
16:17	2	2	GPE0_STS	GPE0_BLK
18:19	2	2	GPE0_EN	GPE0_BLK
20:23	4	4	SETUP_DATA	V-ACPI internal data register.
24:31		8	Unused	Reserved For Future V-ACPI Implementations

The V-ACPI Registers

4.1. PM1a_EVT_BLK – PM Event Block 1a

- 32-bit Aligned
- Contains 16-bit PM1a_STS and PM1a _EN registers.
- Bit assignments can be split between PM1a and PM1b.

4.1.1. PM1a_STS – PM1a Status Register

- 16 bit register containing Fixed Feature status bits.
- Status bits set by ACPI hardware subsystem as a result of an event.
- Status bits are “sticky” (A write of a one (1) to a given bit location will reset the bit.)

Bit	Name	Description
15	WAKE_STS	**Wake Status.** Set when system was in sleep state and an **enabled** wakeup occurs.
12:14	Reserved	
11	PWRBTNOR_STS	**Power Button Override Status.** This bit changes to 1 when the power button is held in for more than 4 seconds.
10	RTC_STS	**Real Time Clock Status.** This bit changes to 1 if an RTC alarm causes a wake up event.
9	SLPBTN_STS	**Sleep Button Status** (Optional) This bit changes to 1 when the sleep button is pressed.

Bit	Name	Description
		If SLPBTN_EN is set, an SCI interrupt is generated.
8	PWRBTN_STS	**Power Button Status** This bit is set when power button is pressed. If PWRBTN_EN is set, an SCI interrupt is asserted.
6:7	Reserved	
5	GBL_STS	**Global Status** The BIOS sets GBL_STS to 1 to release its global lock and return control to the ACPI OS. At the same time GBL_STS is set, the BIOS generates an SCI.
4	BM_STATUS	**Bus Master Status.** This bit changes to 1 whenever a bus master requests the bus.
1:3	Reserved	
0	TMR_STS	**ACPI Timer Status**. This bit changes to 1 whenever bit 23 of the ACPI timer changes state. The ACPI OS is responsible for clearing TMR_STS. If TMR_EN is also set then a SCI interrupt is asserted.

The PM1a_STS Register

4.1.2. PM1a_EN - Enables

- 16 bit register contain Fixed Feature SCI Generation Enable bits.
- These bits enable a SCI interrupt when any of the corresponding status bits is set.

Bit	Name	Description
11:15		Reserved
10	RTC_EN	**Real Time Clock Enable** If set, an SCI is asserted when RTC_STS changes to 1.
9	SLPBTN_EN	**Sleep Button Enable** (Optional) If set, an SCI is asserted when SLPBTN_STS changes to 1.
8	PWRBTN_EN	**Power Button Enable** If set, an SCI is asserted when PWRBTN_STS changes to 1.
6:7	Reserved	
5	GBL_EN	**Global Lock Enable** If set, writing a 1 to GBL_STS causes an SCI to be asserted.
1:4	Reserved	
0	TMR_EN	**ACPI Timer Enable.** If set, an SCI is asserted when bit 23 of the ACPI timer changes state.

The PM1a_EN Register..

4.2. PM1a_CNT_BLK

- Contains PM1a_CNT
- 32 bit aligned

4.2.1. PM1a_CNT - Fixed Feature Control Bits

- 16 bits used
- Various power management enables and communication flags between BIOS and the ACPI OS.

Bit	Name	Description
14:15		Reserved
13	SLP_EN	Setting this bit causes the system to enter the sleep state defined by SLP_TYPx. *Write only. Reads always return a zero.*
10:12	SLP_TYPx	Defines the type of sleep state the system enters when SLP_EN is set.
3:9	Reserved	
2	GBL_RLS	Write only. Used by ACPI OS to raise an event to the BIOS software (SMI). Used by ACPI driver to indicate a release of the global lock and the setting of the pending bit in the FACS table.
1	BM_RLD	When set allows a bus master request to cause a transition from the C3 to the C0 state
0	SCI_EN	Selects whether power management events are SCI or SMI. *Set by hardware based on an ACPI_ENABLE/ACPI_DISABLE written to the SMI_CMD port.*

The PM1a_EN Fixed Feature Enable Register

4.3. P_BLK - Processor Register Block

- 32 bit aligned
- One register block per processor.
- Contains P_CNT, P_LVL2 and P_LVL3

4.3.1. P_CNT - Processor Control

- 32-bit register
- Controls process duty cycle via CPU clock throttling.
- DUTY_WIDTH=3 (can be widened)
- DUTY_OFFSET=0

Bit	Name	Description	
5:31	Reserved	Always zero	
4	THT_EN	Enables throttling of the clock based on the CLK_VAL field.	
3	Reserved	Always zero	
0:2	CLK_VAL	Clock throttling value	
		Value	CPU Duty Cycle
		000	Reserved
		001	12.5%
		010	25%
		011	37.5%
		100	50%
		101	62.5%
		110	75%
		111	87.5%

4.3.2. P_LVL2 - Enter C2 Power State

- 8 bits, read only.
- A read of this register causes the processor to enter C2 power state.
- Reads of P_LVL2 return 0, writes have no effect.

4.3.3. P_LVL3 - Enter C3 Power State

- 8 bits, read only.
- A read of this register causes the processor to enter C3 power state.
- Reads of P_LVL3 return 0, writes have no effect.

4.4. PM2_CNT_BLK

4.4.1. PM2_CNT -

This register is not supported

4.5. SMI_CMD – OS/BIOS Requests

Interpret and process the ACPI commands (defined in Fixed ACPI Description Table):

- 0x01 - ACPI_ENABLE
- 0x02 - ACPI_DISABLE
- 0x03 - S4BIOS_REQ (optional)

5. Generic Address Space

5.1. GPE0_BLK

The GPE0_BLK registers enable system software to configure GPIO (General Purpose I/O) pins to generate SCI interrupts. GPE0_BLK is a 32-bit block aligned on a four byte boundary. It contains two 16-bit registers, GPE0_STS and GPE0_EN, each of which must be configured by the BIOS POST. In order for a GPE0_STS bit to generate an SCI, the corresponding enable bit in GPE0_EN must be set.

5.1.1. GPE0_STS

16-bit, read/write.
GPE0_STS contains the general purpose status event bits. Each bit is set by an external event and cleared by a write of a one to that bit. The GPE0_STS bits are mapped to specific, chipset-resident GPIO signals using the SETUP_IDX and SETUP_DATA registers described later.

Bit	Name	Description
15	OEM_GPE_S15	OEM-Defined
14	OEM_GPE_S14	OEM-Defined
13	OEM_GPE_S13	OEM-Defined
12	OEM_GPE_S12	OEM-Defined
11	OEM_GPE_S11	OEM-Defined
10	OEM_GPE_S10	OEM-Defined
9	OEM_GPE_S09	OEM-Defined
8	OEM_GPE_S08	OEM-Defined
7	OEM_GPE_S07	OEM-Defined
6[1]	LID_STS	Set when lid state changes. If LID_EN is set, a SCI interrupt is asserted. Reset by writing a 1 to this bit. This it GPIO6 on the 5520.
5	OEM_GPE_S05	OEM-Defined
4	OEM_GPE_S04	OEM-Defined
3	OEM_GPE_S03	OEM-Defined
2	OEM_GPE_S02	OEM-Defined
1	OEM_GPE_S01	OEM-Defined
0	OEM_GPE_S00	OEM-Defined

The GPE0_STS Register

5.1.2. GPE0_EN

16 bits
These are the enables for the GPE0_STS bits. When set enables a SCI to be generated when the corresponding GPE0_STS bit is set.

Bit	Name	Description
15	OEM_GPE_E15	OEM-Defined
14	OEM_GPE_E14	OEM-Defined
13	OEM_GPE_E13	OEM-Defined
12	OEM_GPE_E12	OEM-Defined
11	OEM_GPE_E11	OEM-Defined

[1] GPIO6 is included in this table for example purposes only. This is a recommended mapping for the lid switch input. In the recommended configuration, bit 6 of GPE0_STS is mapped to GPIO6 at boot time via SETUP_IDX and SETUP_DATA. Similarly, the lid switch input is routed to GPIO6 in hardware.

Bit	Name	Description
10	OEM_GPE_E10	OEM-Defined
9	OEM_GPE_E09	OEM-Defined
8	OEM_GPE_E08	OEM-Defined
7	OEM_GPE_E07	OEM-Defined
6	LID_EN	Enables LID_STS to generate a SCI when set.
5	OEM_GPE_E05	OEM-Defined
4	OEM_GPE_E04	OEM-Defined
3	OEM_GPE_E03	OEM-Defined
2	OEM_GPE_E02	OEM-Defined
1	OEM_GPE_E01	OEM-Defined
0	OEM_GPE_E00	OEM-Defined

The GPE0_EN Register

5.2. PM_TMR - Power Management timer

- 32-bit aligned.
- LSB 24 bits are a continuous, free-running timer.
- On overflow, TMR_STATUS is set. If TMR_EN is set, TMR_STATUS generates an SCI.
- Exists in physical hardware on the Cx5520/Cx5530 PCI-ISA controllers.

5.3. TEST/SETUP Registers

Cyrix' VSA code provides two additional registers, SETUP_IDX and SETUP_DATA for the purpose of configuring the V-ACPI chipset. Specifically, this pair of registers enables system software to map GPIO pins on the Cx5520/5520 to PM1_STS and GPE0_STS register bits.

5.3.1. SETUP_IDX – Setup Index Register (R/W)

SETUP_IDX is a 16-bit register that references an internal setting in the VSA.
A read of SETUP_IDX returns the last value written to SETUP_IDX.
A write of SETUP_IDX selects the index for a corresponding write to SETUP_DATA.
Writes of any undefined index values to SETUP_IDX are ignored.
If the current value of SETUP_IDX is invalid, a read of SETUP_DATA returns 0.

Index	Operation
0x00	No Operation
0x10	Configure GPIO-0 to PM1A_STS or GPE0_STS bits
0x11	Configure GPIO-1 to PM1A_STS or GPE0_STS bits
0x12	Configure GPIO-2 to PM1A_STS or GPE0_STS bits
0x13	Configure GPIO-3 to PM1A_STS or GPE0_STS bits
0x14	Configure GPIO-4 to PM1A_STS or GPE0_STS bits
0x15	Configure GPIO-5 to PM1A_STS or GPE0_STS bits
0x16	Configure GPIO-6 to PM1A_STS or GPE0_STS bits
0x17	Configure GPIO-7 to PM1A_STS or GPE0_STS bits
0x40	Generate GBL_STS – Sets the GLB_STS bit and generate a SCI to the OS
0x41	Configure IRQ to be used for SCI

5.3.2. SETUP_DATA – Setup Data Register (R/W)

During a read operation, SETUP_DATA returns the value of the internal setting specified by the current value in SETUP_IDX.

5.3.2.1. GPIO Mapping

For GPIO mapping, the following values are used:

0000zyxx where xx is the bit in either the PM1A_STS or GPE0_STS register, y is the edge to be used and z is a runtime/wake indicator

xx value	Function
0x00	No mapping – Do not use this GPIO pin
0x08	Assign GPIOx to PWRBTN_STS bit in PM1A_STS
0x09	Assign GPIOx to SLPBTN_STS in PM1A_STS
0x10	Assign GPIOx to bit 0 in GPE0_STS register
0x11	Assign GPIOx to bit 1 in GPE0_STS register
0x12	Assign GPIOx to bit 2 in GPE0_STS register
0x13	Assign GPIOx to bit 3 in GPE0_STS register
0x14	Assign GPIOx to bit 4 in GPE0_STS register
0x15	Assign GPIOx to bit 5 in GPE0_STS register
0x16	Assign GPIOx to bit 6 in GPE0_STS register
0x17	Assign GPIOx to bit 7 in GPE0_STS register
0x18	Assign GPIOx to bit 8 in GPE0_STS register
0x19	Assign GPIOx to bit 9 in GPE0_STS register
0x1A	Assign GPIOx to bit 10 in GPE0_STS register
0x1B	Assign GPIOx to bit 11 in GPE0_STS register
0x1C	Assign GPIOx to bit 12 in GPE0_STS register
0x1D	Assign GPIOx to bit 13 in GPE0_STS register
0x1E	Assign GPIOx to bit 15 in GPE0_STS register
0x1F	Assign GPIOx to bit 15 in GPE0_STS register

y value	Function
0x0	Rising Edge
0x1	Falling Edge

Z value	Function
0x0	Run time event
0x1	Wake Event

Special considerations for GPIO6 input.

y value	Function
0x0	GPIO6
0x2	LID Switch – STS is set on either edge

Note: GPIO6 provides additional hardware that enables the chipset to generate an SMI on both the rising and falling edges of the input signal. Therefore, the y value for GPIO6 is set to 010b (2d) if an SMI is required on either a rising or falling edge.

5.3.2.2. Mapping SCI interrupt

When mapping the SCI interrupt SETUP_ID contains the number of the IRQ to be used for the SCI. Valid values are 3-7, 9-12, and 14-15. Invalid values will not change the assignment of the SCI IRQ. The default value for the SCI IRQ is 9.

6. V-ACPI Theory Of Operation

V-ACPI relies on the following concepts:

6.1. Fundamental V-ACPI Concepts

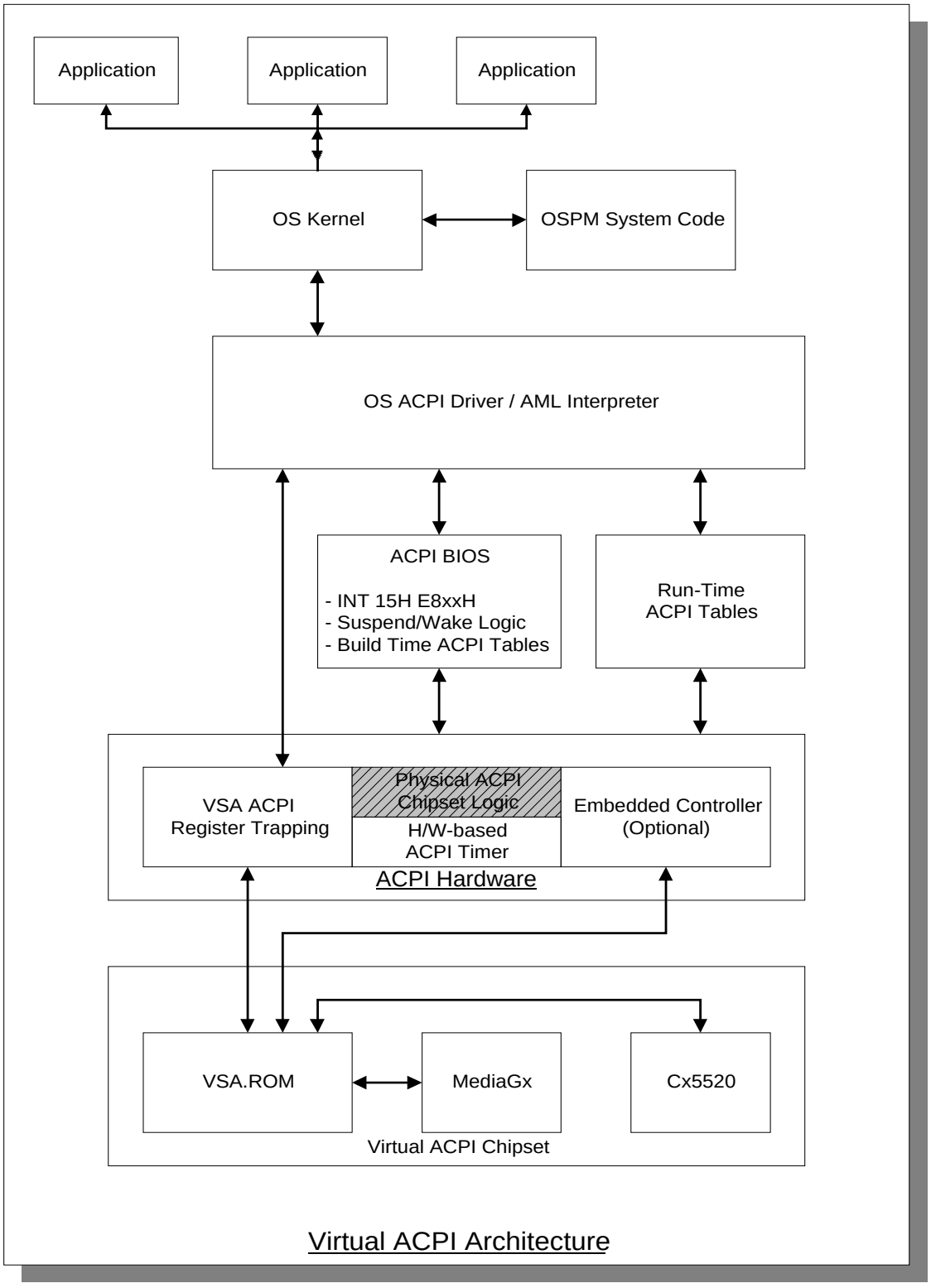
I/O trapping: The chipset is configured to generate an SMI when a particular I/O or memory range is accessed by a foreground process (e.g. Operating System, OS Device Driver). For example, the Fixed Feature Space Registers are located in the I/O range AC00H-AC1FH. Any read or write to registers within this range causes an SMI to occur. The VSA SMI handler is responsible for servicing each V-ACPI SMI interrupt.

Synthetic IRQ: The VSA SMI handler can generate a hardware IRQ by writing certain Cx5520/5530 registers. The IRQ occurs and is serviced by the operating system upon resume from the SMI handler. The capability to generate IRQs in such a manner is useful for responding to ACPI events that ultimately result in an SCI being generated. For example, the overflow of the free-running ACPI timer causes the Cx5520/5530 to generate an SMI. In this case, the SMI handler forces an SCI to occur if TMR_EN is set in PM1a_EN.

H/W ACPI Timer: To eliminate unnecessary latencies, the ACPI timer exists in physical hardware. Reading the current value in the ACPI timer does not cause an SMI to occur.

6.2. System Architecture

The following diagram shows the components in a compliant V-ACPI Cx5520/5530 system.



Virtual ACPI Architecture

6.3. Cx5520/5530 Specific Implementation Details

This section explains the flow of execution for various ACPI events.

6.3.1. Setting CPU Power (Cx) States:

Based on user activity, the CPU places the CPU in state C1, C2 or C3

6.3.1.1. CPU State C1

- The OS issues a HLT instruction in its idle loop.
- When user activity occurs (e.g. interrupt), the CPU automatically returns to state C0.

6.3.1.2. CPU State C2

- The OS reads P_LVL2 to change the CPU state to C2.
- A read of P_LVL2 causes the Cx5520/5530 to generate an SMI.
- V-ACPI code resident in SM-RAM changes the CPU state to C2.
- V-ACPI code configures the chipset to wake the CPU when activity occurs.
- Activity causes the CPU to RSM from SMM and return to the OS.

6.3.1.3. CPU State C3

- Identical to C2 except bus master activity wakes the CPU if BM_RLD is set and a bus mastering device attempts to gain access to memory.

6.3.2. Setting System Power (Sx) States:

6.3.2.1. Sleep State S1

- The OS programs the PM1_CNT register with the S1 sleep type in the _S1 object.
- The OS' write to PM1_CNT causes an SMI to occur.
- The V-ACPI handler records the current sleep type and examines SLP_ENx in PM1_CNT.
- If SLP_ENx is not set, V-ACPI resumes to the OS.
- If it hasn't already done so, the OS sets the SLP_ENx bit in the PM1_CNT register.
- The write to PM1_CNT causes an SMI to occur.
- The V-ACPI handler finds SLP_ENx set in PM1_CNT and changes the system context to S1.
- V-ACPI programs the Cx5520/5530 to wake from S1.
- When user activity occurs, the Cx5520/5530 wakes from S1 and resumes to the OS.

6.3.2.2. Sleep State S2

- The OS programs the PM1_CNT register with the S2 sleep type in the _S2 object.
- The write to PM1_CNT causes an SMI to occur.
- The V-ACPI handler records the current sleep type and examines SLP_ENx in PM1_CNT.
- If SLP_ENx is not set, V-ACPI resumes to the OS.
- The OS issues a WBINVD instruction to flush the processor cache.
- If it hasn't already done so, the OS sets the SLP_Enx bit in the PM1_CNT register.
- The write to PM1_CNT causes an SMI to occur.
- The V-ACPI handler finds SLP_ENx set in PM1_CNT.
- V-ACPI stores the Wake type in non-volatile RAM.
- V-ACPI configures the hardware to enter state S2 (CPU, cache, device context lost)
- Upon waking, V-ACPI locates the ACPI BIOS wake vector and resumes to it.

6.3.2.3. Sleep State S3

- V-ACPI does not support sleep state S3.

6.3.2.4. Sleep State S4 (OS Initiated)

- The OS initiates S4 state by writing the SLP_TYPx and SLP_EN bits in PM1_CNT.
- The write to PM1_CNT causes an SMI.
- V-ACPI stores the sleep type in non-volatile RAM.
- V-ACPI code removes power from all devices.
- Upon reset, the BIOS restores memory controller context from NV-RAM.
- During a resume from S4 (determined by SLP_TYPx in NV-RAM), the BIOS updates the system hardware signature and performs a normal boot. The OS is responsible for restoring device context.

6.3.2.5. Sleep State S4BIOS

- The OS writes S4BIOS_REQ to the SMI_CMD port.
- The write to SMI_CMD causes an SMI.
- V-ACPI code saves its context to a disk file or partition and removes power from all devices.
- Upon system reset, if NV-RAM contains SLP_TYPE_S4BIOS, the BIOS restores memory controller context from NV-RAM.
- During a resume from S4BIOS, the BIOS initializes boot devices and jumps to the OS wake vector.

6.3.3. ACPI Timer SCI Generation:

The free-running hardware ACPI timer in the Cx5520/5530 controllers generates an SMI each time it overflows (approximately every four seconds). If the TMR_EN bit is set, V-ACPI will generate a corresponding, synthetic SCI that occurs upon each resume from a timer SMI.

6.3.4. GPIO Event Generation:

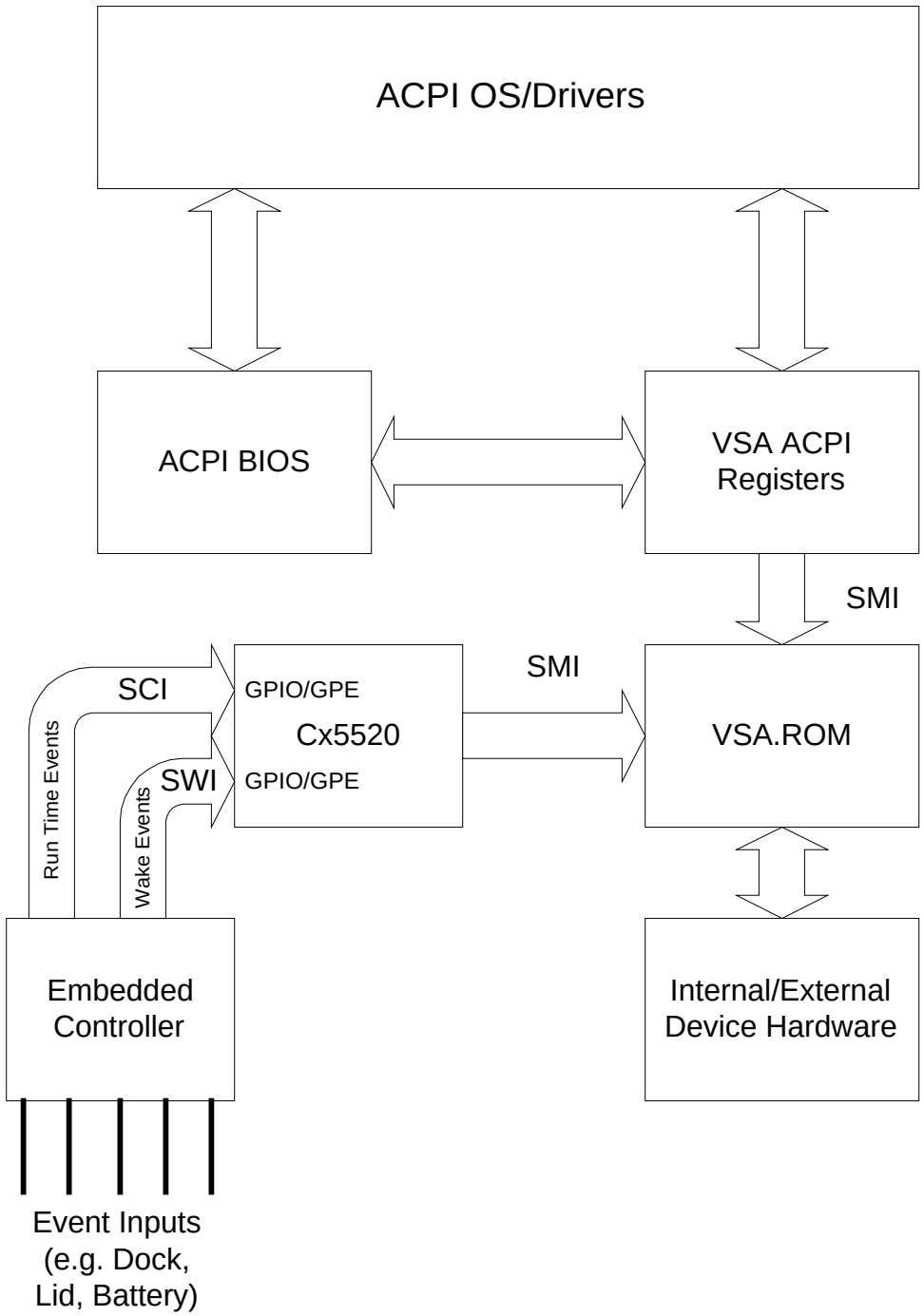
The BIOS programs certain GPIO pins to respond to events using the SETUP_IDX and SETUP_DATA registers described previously. GPIO pins that have been configured as GPE inputs cause an SMI to occur. V-ACPI sets the appropriate GPE0_STS bit(s) for each GPE pin event. If the enable bit for a particular GPE input is set, V-ACPI generates a synthetic SCI for that GPE.

6.3.4.1. Optional Embedded Controller Support

Embedded controller support adds functionality to portables in areas such as thermal monitoring, hotkey support and system power state control. To support an embedded controller in an ACPI design, the chipset must provide two GPIO inputs. In legacy mode, only one input (EXTSMI) is supported. In ACPI mode, the system requires two inputs. The first (EXTSCI) is asserted by the embedded controller to report a non-wake related ACPI event, the meaning of which is determined by the BIOS AML code. The second input (SWI) is used by the embedded controller to report wake events to the ACPI chipset.

In a V-ACPI system, each GPE0_STS or GPE0_EN bit can be assigned to any GPIO pin on the chipset. Additionally, V-ACPI provides the SETUP_IDX and SETUP_DATA registers to the BIOS for the purpose of initializing the meaning of the GPE inputs. Each input has programmable edge triggering, and can be configured as either a wake or run-time event input.

The following diagram displays the block diagram architecture of a V-ACPI system with embedded controller support.



ACPI Model #2 -- Portable w/ACPI Embedded Controller