

第五章 時脈與電源管理

時脈與電源管理員控制每個模組的時脈頻率，以及管理不同的電源管理操作模式的轉換，藉此達到最佳的計算效能與電源消耗。

5.1 時脈管理員簡介

時脈與電源管理員提供每個週邊單元固定的時脈。大部分的週邊時脈可使用時脈致能暫存器(Clock Enable Register--CKEN)或者經由週邊的控制暫存器的位元來關閉。關閉任何沒有在使用的單元的時脈，以降低電源的消耗。時脈與電源管理員也提供可程式的頻率時脈供 LCD 控制器、記憶體控制器與 CPU 使用。這些時脈之間彼此相依，因為它們來自同一個內部相位鎖定迴路(Phase Locked Loop--PLL)所產生的時脈來源。根據以下的步驟(關於 L、M、N 的說明請參考，5.6.1 節「核心時脈組態暫存器」)來設定 PLL 的頻率：

1. 決定出最快的同步記憶體(SDRAM)頻率需求。
2. 若 SDRAM 的頻率低於 99.5MHz，則記憶體頻率必須為 SDRAM 頻率的兩倍，而且記憶體控制器內的 SDRAM 時脈倍率必須設定為 2。若 SDRAM 頻率為 99.5MHz，則記憶體頻率則必須與 SDRAM 頻率相同。
3. 變更核心時脈組態暫存器(Core Clock Configuration Register)內的 L 值，可將外部同步記憶體頻率設定為 99.5MHz(L = 0x1B)、118.0MHz(L = 0x20)、132.7MHz(L = 0x24)、147.5MHz(L = 0x28)、165.9MHz(L = 0x2D)等的近似值，若 SDRAM 時脈比率(SDRAM clock ratio)為 2，則頻率減半。
4. 執行模式(Run mode)的核心頻率。當應用程式的程式碼偶爾會從外部記憶體做抓取動作時，使用此模式處理正常的運算。變更核心時脈組態暫存器內的 M 值可以變更核心頻率，核心頻率通常為記憶體頻率的 1 倍、2 倍或 4 倍。
5. 加速模式(Turbo mode)的核心頻率。當應用程式的程式碼整個都在快取記憶體(caches)內時，可以使用此模式處理運算。使用此模式時，任何從外部記憶體抓取程式碼的動作都會降低核心的運算效能。變更核心時脈組態暫存器內的 N 值可以變更核心頻率。核心頻率通常為執行模式(Run mode)的 1.0、1.5、2.0 或 3.0 倍。
6. 為變更後的核心頻率會設定新的記憶體控制器與 LCD 控制器的組態，並且進入頻率改變程序(Frequency Change Sequence)(5.4.7 節「頻率改變程序」有相關的說明)。

注意：並非所有的頻率組合都是有效的。關於有效的組合請參考 5.3.3 節「核心相位鎖定迴路」。

5.2 電源管理員簡介

時脈與電源管理員可以下列三種之一的方法將應用處理器重置：

- 硬體重置(Hardware Reset) 。觸發 nRESET 將被視為一個不可遮蔽的完全重置。使用時機為電源開機或是當系統沒有任何資訊需要被保留時使用。
- 看門狗重置(Watchdog Reset) 。利用看門狗計時器來重置整個系統，時脈與電源管理除外。通常可以利用此種重置來當作程式碼看門狗。如果當程式碼無法完成一個指定的程序時，應用處理器會假設發生一個嚴重的系統錯誤，並且觸發看門狗重置。
- GPIO 重置(GPIO Reset) 。透過 GPIO 交替功能暫存器(GPIO alternate function register)來致能 GPIO 重置。提供另一種硬體重置的選擇性，並且保留記憶體控制器暫存器的狀態，部分時脈、電源管理員與 RTC 的狀態。

時脈與電源管理員也會控制應用處理器上，所有的低電源模式或特殊時脈模式的進入與離開。這些模式為：

- 加速模式：核心以其最高的頻率來執行。在此模式下，盡量減少外部記憶體的存取，因為核心必須等待外部記憶體的存取。
- 執行模式：核心以正常頻率來執行。在此模式下，假設核心會對外部記憶體頻繁地存取，因此可以較慢的速度來執行，以達到最理想的電源與效能的平衡，使用者可以自行衡量。
- 閒置模式：除了核心沒有供應時脈之外，系統其餘的部分依然正常運作。當核心必須等待外部系統，短暫地閒置並且停止運算時，可以使用此種模式。
- 睡眠模式：應用處理器會處於最低的電源消耗狀態，同時維持 I/O 狀態、RTC 及時脈與電源管理員的狀態。如果要從睡眠模式中喚醒，系統必須重新開機，但大部分的內部狀態會消失。在睡眠模式時，核心電源必須接地，以防止電流流失。

時脈與電源管理員在頻率改變程序時，也控制應用處理器在此期間的動作。頻率改變程序是一種將核心頻率(執行與加速模式)與記憶體頻率改變的程序，將核心時脈組態暫存器(Core Clock Configuration Register)內的值，從先前儲存的值改變為新的值。此程序必須花費一些時間來完成，主要是因為 PLL 重新鎖定需要一些時間，但它也允許動態的頻率改變，而不用與外部記憶體的完整性來妥協。任何依賴核心或記憶體控制器的週邊，必須設定成可以承受任何的資料流中斷。

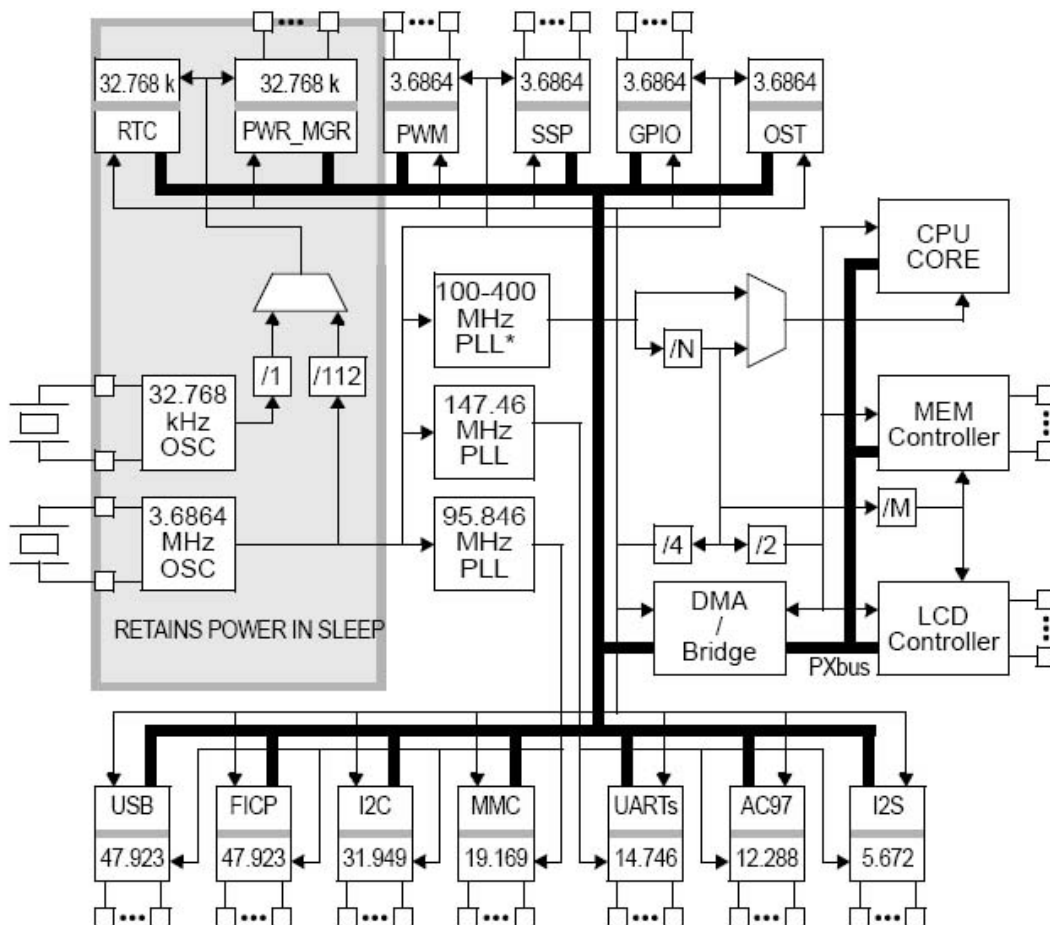
5.3 時脈管理員

應用處理器的時脈系統包含 5 個主要的時脈來源：

- 32.768kHz 振盪器
- 3.6864MHz 振盪器
- 可程式頻率核心 PLL
- 95.85MHz 固定頻率週邊 PLL(Fixed Frequency Peripheral PLL)
- 147.46MHz 固定頻率 PLL(Fixed Frequency PLL)

時脈管理員也透過時脈閘控(clock gating)的方式來減少電源的損耗。圖 5-1 依照功能來顯示時脈的配送網。「L」代表核心 PLL。

如圖 5-1 所示，PXbus 為一個介於核心、DMA/Bridge、LCD 控制器與記憶體控制器之間的內部匯流排。此匯流排執行時脈為執行模式頻率的 1/2。為了達到最佳效能，PXbus 的執行時脈越快越好。舉例說明，若目標核心頻率為 200MHz，則會以 200MHz 的執行模式替代 200MHz 的加速模式，匯流排便以 100MHz 來運作。提升 PXbus 頻率可幫助減少存取不可快取的記憶體所產生的延遲。



* For the PXA210 application processor: 100-200 MHz.
For the PXA250 application processor: 100-400 MHz.

圖 5-1 時脈管理員區塊圖

5.3.1 32.768kHz 振盪器

32.768kHz 振盪器為低耗電與低頻率的振盪器，提供 RTC 和電源管理員時脈。此振盪器在硬體重置後關閉，RTC 和電源管理員便無法使用，所以用 3.6864MHz 振盪器取代此振盪器。軟體寫入振盪器組態暫存器 (Oscillator Configuration Register) 內的振盪器啟動位元 (On Bit)，來啟動 32.768kHz。等待時脈穩定之後，再設定組態 RTC 和電源管理員來使用 32.768kHz 振盪器。

是否要使用 32.768kHz 振盪器是可以選擇的，此振盪器在睡眠模式期間，提供最少的電源消耗。在電源敏感度較低的應用程式，為了節省成本，可以關閉在 OSCC (Oscillator Configuration Register) 裡的 32.768kHz 振盪器，並且保持該腳位浮接 (不需要使用外部晶體)。如果在系統內不使用 32.768kHz 振盪器，RTC 和電源管理員的頻率會來自 3.6864MHz 除以 112 (32.914kHz) 來產生。在睡眠模式時，若 3.6864MHz 振盪器持續致能時，則會消耗數百微安培的額外電源。振盪器電源關閉致能位元 (Oscillator Power Down Enable - OPDE) 將決定在睡眠模式時是否致能 3.6864MHz 振盪器，相關資料請參考 5.5.2 節「電源管理通用組態暫存器」。不需要外部電容。

5.3.2 3.6864MHz 振盪器

3.6864MHz 振盪器提供主要的時脈來源給應用處理器使用。On-chip PLL 頻率乘法器 (frequency multiplier)、同步序列埠 (Synchronous Serial Port - SSP)、脈衝寬度調變器 (Pulse Width Modulator - PWM) 和作業系統計時器 (Operating System Timer - OST) 使用 3.6864MHz 振盪器作為參考頻率。在硬體重置之後，3.6864MHz 振盪器也會提供頻率給 RTC 和 PM (電源管理員)。然後使用者可以致能 32.768kHz 振盪器，待 32.768kHz 振盪器的頻率穩定後，32.768kHz 振盪器將會驅動 RTC 和 PM。當 32.768kHz 振盪器為致能而且穩定時 (在 OSCC 內的 OON 位元和 OOK 位元皆設定)，在睡眠期間可以經由設定 OPDE 位元 (請參考 5.5.2 節) 來關閉 3.6864MHz 振盪器。關於更多內容請參考 5.6.3 節。不需要外部電容。

5.3.3 核心相位鎖定迴路 (Core Phase Locked Loop)

核心 PLL 提供時脈來源給 CPU 核心、記憶體控制器、LCD 控制器和 DMA

控制器。核心 PLL 使用 3.6864MHz 振盪器作為頻率參考，乘以下列變數來得到其頻率：

- L：晶體(crystal)頻率對記憶體頻率的乘數，設定為 27、32、36、40 或 45。
- M：記憶體頻率對執行模式(Run mode)的頻率的乘數，設定為 1 或 2。
- N：執行模式(Run mode)的頻率對加速模式(Turbo mode)的頻率的乘數，設定為 1.0、1.5、2.0 或 3.0。

關於輸出頻率選擇請參考表 5-1「3.6864MHz 晶體的核心 PLL 輸出頻率」。關於 L、M 和 N 的程式化資料請參考 5.6.1 節。參考 5.6.1 節「核心時脈組態暫存器」的十六進位設定。

當應用處理器正在運算時，不要選擇不在應用處理器支援範圍的電壓與封裝類型之頻率組合。

SDCLK 不可大於 100MHz。若 MEMCLK 大於 100MHz，則在記憶體控制器裡，SDCLK 和 MEMCLK 的比率必須設定為 1:2。

表 5-1 3.6864MHz 晶體的核心 PLL 輸出頻率

| L | M | “N”所對應的加速模式頻率(MHz)與 核心時脈組態暫存器(CCCR[15:0])內 的“N”值 | | | | PXbus 頻率 | MEM、LCD 頻率(MHz) | SDRAM 最高頻率 |
|----|---|--|----------------|----------------|----------------|-------------|--------------------|---------------|
| | | 1.00 (Run) | 1.50 | 2.00 | 3.00 | | | |
| 27 | 1 | 99.5 @.85v | — | 199.1 @1.0v | 298.6 @1.1v | 50 | 99.5 | 99.5 |
| 32 | 1 | 118.0 @1.0v | — | 235.9 @1.1v | 353.9 @1.3v | 59 | 118.0 | 59.0 |
| 36 | 1 | 132.7 @1.0v | — | 265.4 @1.1v | 398.1 @1.3v | 66 | 132.7 | 66 |
| 40 | 1 | 147.5 @1.0v | — | 294.9 @1.1v | — | 74 | 147.5 | 74 |
| 45 | 1 | 165.9 @1.0v | — | 331.8 @1.3v | — | 83 | 165.9 | 83 |
| 27 | 2 | 199.1 @1.0v | 298.6 @1.1v | 398.1 @1.3v | — | 99.5 | 99.5 | 99.5 |
| 32 | 2 | 235.9 @1.1v | — | — | — | 118 | 118.0 | 59.0 |
| 36 | 2 | 265.4 @1.1v | — | — | — | 132.7 | 132.7 | 66 |

| | | | | | | | | |
|----|---|----------------|---|---|---|-------|-------|----|
| 40 | 2 | 294.9 @1.1v | — | — | — | 147.5 | 147.5 | 74 |
| 45 | 2 | 331.9 @1.3v | — | — | — | 165.9 | 165.9 | 83 |

5.3.4 95.85MHz 週邊相位鎖定迴路

95.85MHz PLL 為許多週邊區塊的外部介面提供時脈來源。這些外部介面需要~48MHz(UDC／USB、FICP)、~33MHz(I2C)與~20MHz(MMC)等頻率。所產生的頻率並沒有完全等於所需的頻率，這是由於所選的晶體與單元之間的缺乏完美的最小公倍數(Least Common Multiple)的關係。所以所選用的時脈頻率會維持在每個單元所能忍受的範圍內。若使用 3.6864MHz 以外的晶體，週邊區塊介面的時脈頻率可能無法產生所需的鮑率(或協定比率(protocol's rate))。

表 5-2 3.6864MHz 產生的 95.85MHz 週邊 PLL 輸出頻率

| 單元名稱 | 正常頻率 | 實際頻率 |
|----------|--------|------------|
| USB(UDC) | 48 MHz | 47.923 MHz |
| FICP | 48 MHz | 47.923 MHz |
| I2C | 33 MHz | 31.949 MHz |
| MMC | 20 MHz | 19.169 MHz |

5.3.5 147.46MHz 週邊相位鎖定迴路

147.46MHz PLL 為許多週邊區塊的外部介面提供時脈來源。這些外部介面需要~14.75MHz(UARTs)、12.288MHz(AC97)與可變頻率(I2S)等頻率。所產生的頻率並沒有完全等於所需的頻率，這是由於所選的晶體與單元之間的缺乏完美的最小公倍數(Least Common Multiple)的關係。所以所選用的時脈頻率會維持在每個單元所能忍受的範圍內。若使用 3.6864MHz 以外的晶體，週邊區塊介面的時脈頻率可能無法產生所需的鮑率(或協定比率(protocol's rate))。

表 5-3 3.6864MHz 晶體產生的 147.46MHz 週邊 PLL 輸出頻率

| 單元名稱 | 正常頻率 | 實際頻率 |
|-------|------------|------------|
| UARTs | 14.746 MHz | 14.746 MHz |
| AC97 | 12.288 MHz | 12.288 MHz |

| | | |
|-----|------------|------------|
| I2S | 146.76 MHz | 147.46 MHz |
|-----|------------|------------|

5.3.6 時脈閘控(Clock Gating)

時脈管理員包含 CKEN 暫存器。此暫存器包含組態位元，可用來關閉個別單元的時脈。當模組沒有被使用時，其組態位元應該關閉。在硬體重置之後，任何沒有使用的模組必須關閉其時脈。若模組處於暫時停止的狀態時，且該單元不具備時脈閘控的功能時，則可以利用 CKEN 暫存器來關閉該單元的時脈。

當一個模組的時脈關閉時，其暫存器依然為可讀寫。而 AC97 則為一個例外，若其時脈已關閉，則完全無法存取。

5.4 重置與電源模式

時脈與電源管理員單元決定應用處理器的重置、電源程序與電源模式。每一種模式的行為在運作中都不相同，且具有特定的進入與離開程序。重置與電源模式為：

- 硬體重置(Hardware Reset)
- 看門狗重置(Watchdog Reset)
- GPIO 重置(GPIO Reset)
- 執行模式(Run Mode)
- 加速模式(Turbo Mode)
- 閒置模式(Idle Mode)
- 頻率改變程序(Frequency Change Sequence)
- 睡眠模式(Sleep Mode)

5.4.1 硬體重置

若要觸發硬體重置讓所有在應用處理器裡的單元重置為已知狀態時，可以觸發 nRESET 腳位。當電源開啟與完全重置時才會需要使用此硬體重置。

5.4.1.1 引發硬體重置

當nRESET腳位被外部來源降低成低電位時，將引發硬體重置。沒有任何方法可以遮蔽或是關閉來自外部腳位對應用處理器的重置。當nRESET被觸發，不論是哪种操作模式，都會引發硬體重置。當nRESET腳位被觸發時，nRESET-OUT腳位也會被觸發。nRESET必須在 t_{DHW_NRESET} 時間內保持低電位，讓系統達到穩定狀態，以及讓重置的訊息傳遞到各個單元。相關細節請參考Intel® PXA250 and PXA210 Application Processors Electrical, Mechanical, and Thermal Specification。

5.4.1.2 硬體重置期間之行為

在硬體重置期間，所有內部暫存器和單位都維持其定義的重置狀態。當nRESET 腳位觸發時，除了 3.6864MHz 振盪器之外，沒有任何東西在應用處理器裡是處於運作狀態的。內部時脈停止且晶片是靜止的。所有腳位會回到它們的重置狀態，且會忽略 nBATT_FAULT 和 nVDD_FAULT 腳位。因為記憶體控制器接收到完全重置的訊息，所有動態記憶體的內容在硬體重置期間會全部遺失。

5.4.1.3 完成硬體重置

為了完成硬體重置，必須移除nRESET腳位的觸發。在nRESET的觸發移除之前，所有的電源供應必須保持穩定一段時間(t_{D_NRESET})。相關細節請參考Intel® PXA250 and PXA210 Application Processors Electrical, Mechanical, and Thermal Specification。在nRESET的觸發移除之後，會發生以下程序：

1. 3.6864MHz 振盪器和內部 PLL 時脈產生器等待達到穩定狀態。
2. nRESET_OUT 腳位觸發移除。
3. 開始一般開機(boot-up)程序。所有應用處理器單元回到它們預先定義的重置狀態。軟體必須檢查重置控制狀態暫存器(Reset Controller Status Register - RCSR)來判定重新開機的原因。

5.4.2 看門狗重置

當軟體無法適當地避免看門狗逾時事件的發生時，就會引發看門狗重置。只有當軟體無法正確地執行以及可能有損壞的資料時，才會產生看門狗重置。在看門狗重置時，除了時脈與電源管理員之外，所有單元都會重置。

5.4.2.1 引發看門狗重置

當 OWER 裡面的看門狗致能位元(Watchdog Enable bit - WE)被設定時，且 OSMR[3]符合作業系統計時計數器(OS Timer Counter)時，才會引發看門狗重置。

在這些情況下，不論之前的運作模式為合，都會引發看門狗重置。看門狗重置會觸發 nRESET_OUT。

5.4.2.2 看門狗重置期間之行為

在看門狗重置期間，除了即時時脈(Real Time Clock) 與部分的時脈以及電源管理員之外，所有的單元會回到其預先定義的重置狀態。除了振盪器腳位之外的所有腳位會呈現重置狀態，而 nBATT_FAULT 與 nVDD_FAULT 腳位會被忽略。在看門狗重置期間，所有的動態記憶體的內容都會遺失，因為記憶體控制器接受到完全重置的訊息。

關於看門狗重置與其它重置期間腳位的狀態，請參考表 4-7「PXA250 應用處理器腳位與信號說明」與表 4-9「PXA210 應用處理器腳位與信號說明」。

5.4.2.3 完成看門狗重置

當 nRESET 腳位觸發時，看門狗重置會立刻回復到硬體重置。此外，看門狗重置的完整程序為：

1. 在經過 t_{DHW_OUT} 之後，看門狗重置的觸發來源會被移除。請參考 *Intel® PXA250 and PXA210 Application Processors Electrical, Mechanical, and Thermal Specification*。
2. 3.6864MHz 振盪器與內部相位鎖定迴路時脈產生器會一直等待達到穩定狀態為止。32.768kHz 振盪器的組態與狀態不會受到看門狗重置的影響。
3. nRESET_OUT 腳位的觸發會移除。*Intel® PXA250 and PXA210 Application Processors Electrical, Mechanical, and Thermal Specification*。
4. 正常開機程序會開始。除了即時時脈裡面的 RTTR、部分的時脈、及電源管理員之外的所有應用處理器單元都會返回其預先定義的重置狀態。軟體必須檢查 RCSR 來判定重新開機的原因。

5.4.3 GPIO 重置

當 GP[1]被適當地設定為重置來源時，而且觸發維持低電位的時間大於 $4 \times N$ 的處理器時脈週期(N 為加速模式時脈乘數，在核心時脈組態暫存器內—請參考 5.6.1 節)時，會引發 GPIO 重置。GPIO 重置時，除了即時時脈與部分時脈與電源管理員，以及記憶體控制器之外的所有的應用處理器單元，都會回復其預先定義且已知的狀態。

5.4.3.1 引發 GPIO 重置

為了使用 GPIO 重置的功能，必須經由 GPIO 控制器來設定。GP[1]腳位必須設定成輸入，而且在 GPIO 控制器內設定為 GPIO 重置功能。GPIO 重置功能為位準感應(level-sensitive)，且不為邊緣觸發(edge-triggered)。當 GPIO 重置功能被設定時，依照以下步驟來確定不會產生假的重置：

1. GP[1]必須設定為輸出，且其資料暫存器設定為 1。
2. 從外部將 GP[1]腳位驅動為高電位狀態。
3. 設定 GP[1]為輸入。
4. 設定 GP[1]的可交替(重置)功能。

GPIO 重置之前的運作模式不會影響 GPIO 重置。當引發 GPIO 重置時，nRESET_OUT 會被觸發。若 GP1 觸發少於 4xN 個週期時，則應用處理器則維持先前的運作模式，否則進入 GPIO 重置。

GPIO 重置在睡眠模式下不會運作，因為所有的 GPIO 可交替功能的輸入腳位都關閉。在睡眠模式期間，外部喚醒來源必須被重導至其中一個致能的 GPIO 喚醒來源(相關細節請參考 5.5.3 節)。

5.4.3.2 GPIO 重置期間之行為

GPIO 重置期間，大部分(非全部)的內部暫存器與處理器會維持在預先定義的重置狀態。而 RTC、時脈、電源管理員(特別註明者除外)與記憶體控制器則為例外。在 GPIO 重置期間，時脈單元會以其先前設定的值繼續運作，因此應用處理器會以相同的時脈組態進入與離開 GPIO 重置除了振盪器與記憶體控制器腳位之外，所有的腳位都會回復到重置狀態，但忽略 nBATT_FAULT 與 nVDD_FAULT 腳位所產生的觸發。

GPIO 重置不會重置記憶體控制組態暫存器(Memory Controller Configuration Register)。若外部記憶體在 GPIO 重置之前有適當地設定，則外部記憶體的內容可以被保留。在 GPIO 重置期間，為了保留 SDRAM 的內容，軟體必須正確地設定記憶體控制組態，而且 GPIO 重置所耗費的時間必須少於 SDRAM 的刷新週期。GPIO 重置所耗費的時間相依於 GPIO 重置之前的 CPU 運作模式。詳細資料請參考第 6 章「記憶體控制器」。

關於 GPIO 重置期間 PXA250 應用處理器全部腳位的狀態請參考表 4-7「PXA250 應用處理器腳位與信號說明」。而 PXA210 應用處理器請參考表 4-9「PXA210 應用處理器腳位與信號說明」。

5.4.3.3 完成 GPIO 重置

當 nRESET 腳位被觸發，GPIO 重置會立即回復到硬體重置。GPIO 重置完成程序如下：

1. 因為內部重置被傳送至 GPIO 控制器與其暫存器，並回到其重置狀態，使 GPIO 重置觸發的來源被移除。
2. nRESET_OUT 觸發移除。
3. 開始正常開機程序。除了即時時脈、部分的時脈、電源管理員、記憶體控制器之外，所有應用處理器單元會回復到預先定義的重置狀態。軟體必須檢查 RCSR 來判定重新開機的原因。

5.4.4 執行模式(Run Mode)

執行模式為應用處理器的一般運作模式。所有的電源供應皆致能，且所有已經致能的功能的時脈都正常運作。在完成電源模式(Power Mode)、電源程序(Power Sequence)或重置(Reset) 程序之後，會進入執行模式。當任何電源模式、電源程序或重置執行模式離開時，就會離開執行模式。

5.4.5 加速模式(Turbo Mode)

當使用者需要最高的處理效能時，加速模式允許應用處理器的核心時脈頻率以最高的頻率來運作。加速模式也允許在不同的頻率之間同步轉換，而不會中斷任何記憶體控制器、LCD 控制器，或是其它週邊裝置。

5.4.5.1 進入加速模式

執行模式的處理器時脈頻率和加速模式的處理器時脈頻率之比例在 CCCR[N]內被設定。CCCR[N]內的值與其它任何其它的適當時脈組態，必須要經由頻率改變程序的步驟來設定。同時改變加速模式以及進入頻率改變程序，也要依照時脈改變程序的步驟來改變。

當軟體在時脈設定暫存器(Clock Config Register - CCLKCFG) (請參考 5.7.1 節)將加速位元設定時，加速模式將會被引發。在軟體設定加速位元之後，CPU 必須等待所有在管線(pipeline)內的指令完成。當指令完成之後，CPU 會繼續運作，並且以比較高的加速模式頻率來運作。

軟體可以在設定或清除 CCLKCFG 裡面的其它位元時，同時設定加速位元。暫存器裡面的其它位元的設定優先權會高於加速位元，所以說在 CPU 進入加速模式之前，會優先執行其它位元的設定模式。當 CPU 離開其它模式時，CPU 會以 CCLKCFG[Turbo]位元的設定狀態為基準，進入執行模式或是加速模式。

不要把協力處理器 14 內之 CCLKCFG 暫存器與應用處理器的時脈與電源管理員內之 CCCR(請參考 5.6.1 節) 混淆在一起。

5.4.5.2 加速模式之行為

除了與處理器時脈頻率相對應的記憶體，以及與處理器時脈頻率相對應的相關週邊，其頻率會隨著 CCCR 內的 N 值(請參考 5.6.1 節)增加之外，應用處理器在加速模式與在執行模式的行為是一樣的。當對外部記憶體的存取機率很少時，適合以加速模式來進行運算。當較高的核心頻率與外部記憶體時脈頻率的比率增加時，會相對應地增加每個對外部記憶體存取的延遲。因此，增加的延遲會降低應用處理器的電源效能。為了達到最佳效能，軟體必須在執行模式時將應用程式載入快取記憶體內，然後在加速模式時執行。

5.4.5.3 離開加速模式

為了離開加速模式，軟體必須清除 CCLKCFG 暫存器內的加速位元。在軟體清除加速位元之後，CPU 必須等待管線(pipeline)內的指令運算完成之後，CPU 便會進入執行模式。

當設定或清除 CCLKCFG 裡面的其它位元時，同時清除加速位元，暫存器裡面的其它位元的設定優先權會高於加速位元，所以 CPU 新的運作模式(執行模式)的程序會在其它位元設定之後開始。

閒置模式、睡眠模式、頻率改變程序以及重置等的處理都會比加速模式優先執行，並且會造成應用處理器離開加速模式。當 CPU 離開任何上述的模式時(閒置模式、睡眠模式、頻率改變程序以及重置)，CPU 會以 CCLKCFG[Turbo]之設定狀態為基礎，進入執行模式或是加速模式。

5.4.6 閒置模式(Idle Mode)

閒置模式允許使用者在處理器停止運算期間，繼續監視 on-chip 與 off-chip

的中斷服務請求，並且停止 CPU 的核心時脈。閒置模式不會改變時脈，因此當中斷發生時，CPU 可以快速地恢復到進入閒置模式之前的狀態。

在閒置模式期間，以下的資源會繼續運作：

- 系統單元模組(即時時脈、作業系統計時器、中斷控制器、通用 I/O，以及時脈與電源管理員)
- 週邊單元模組(DMA 控制器、LCD 控制器，以及所有其它的週邊單元)
- 記憶體控制器

5.4.6.1 進入閒置模式

在閒置模式期間，會停止供應 CPU 的核心時脈。所有重要的應用程式必須完成且結束。當週邊需要 CPU 的運算時，週邊必須在進入閒置模式之前設定成可以對 CPU 產生中斷。軟體會對 PWRMODE[M](請參考 5.7.2 節)做選擇後，並且進入閒置模式。當中斷發生時，CPU 會立即終止閒置模式，並且繼續閒置前的正常運算。在軟體選擇完閒置模式之後，CPU 會等待，直到管線(pipeline)內所有的指令皆完成。當指令完成後，CPU 的時脈會停止，並而開始進入閒置模式。在閒置模式下，中斷源會被當作是為喚醒(wake-up)CPU 的來源。

5.4.6.2 閒置模式之行為

在閒置模式下 CPU 的時脈會停止，但應用處理器的其餘的部分依然正常地運作。舉例來說，LCD 控制器可以用記憶體內的同一個訊框資料來刷新螢幕內容。

當中斷控制器控制暫存器 ICCR[DIM]被清除時，任何一個致能的中斷皆可以喚醒應用處理器。當 ICCR[DIM]被設定，只有未遮蔽的中斷可以喚醒應用處理器。

致能的中斷(Enabled interrupt)為單元層次所允許的中斷。遮蔽的中斷(Masked interrupt)為避免中斷核心的中斷，並且在中斷控制器之遮蔽暫存器(Interrupt Controller Mask Register)裡設定。

5.4.6.3 離開閒置模式

若有任何重置發生時，則會離開閒置模式。進入與離開重置的程序的優先等級皆高於閒置模式。當重置離開程序完成之後，CPU 不會繼續維持在閒置模式。如果看門狗計時器被致能時，軟體必須在設定閒置模式之前先設定看門狗符合暫

存器(Watchdog Match Register)，以確定在看門狗重置觸發發生之前，另一個中斷可以使應用處理器離開閒置模式。使用 RTC 警報信號或另一個 OS 計時器通道來達到此目的。

任何致能的中斷都可以使 CPU 離開閒置模式。當 ICCR[DIM]被清除時，在閒置模式期間，中斷控制器之遮蔽暫存器(ICMR)的設定會被忽略。這表示不一定要取消遮蔽才能利用中斷來離開閒置模式。離開閒置模式的程序如下：

1. 一個有效的、致能的中斷觸發。
2. CPU 時脈重新啟動，然後 CPU 依照 CCLKCFG[Turbo]內的設定恢復運算。

當 nBATT_FAULT 或 nVDD_FAULT 腳位被觸發時也會離開閒置模式。當這兩個腳位的任何一個腳位被觸發時，閒置模式的離開程序如下：

1. nBATT_FAULT 或 nVDD_FAULT 腳位被觸發。
2. 若電源管理控制暫存器(Power Manager Control Register - PMCR)內的不正確資料放棄致能(Imprecise Data Abort Enable - IDAE)位元被清除(不建議如此使用)，則應用處理器會立即進入睡眠模式。
3. 若 IDAE 位元被設定，對時脈模組而言，nBATT_FAULT 或 vDD_FAULT 的觸發是有效的，應用處理器會依照正常的中斷觸發程序來離開閒置模式。然後軟體必須關閉系統，並且進入睡眠模式。更多的相關細節請參考 5.4.8.3 節「進入睡眠模式」。

5.4.7 頻率改變程序(Frequency Change Sequence)

頻率改變程序最主要的目的就是改變處理器時脈的頻率。在頻率改變程序的期間，CPU、記憶體控制器、LCD 控制器和 DMA 等的時脈會停止，而其它的週邊會繼續維持運作。此程序是使用來改變初始開機(initial boot-up)時預設狀態的頻率，或者也可以用來當做節省電力的功能，當軟體改變頻率後，以最低需求的頻率來執行運算。

5.4.7.1 頻率改變程序的先前準備

在軟體初始化頻率改變程序之前，必須完成以下的步驟：

1. 在頻率改變程序期間，適當地設定記憶體控制器的組態來確保 SDRAM 內的資料。記憶體控制器的刷新計時器必須設定為目前的工作頻率與未來工作頻率兩者之間取最慢者的刷新時間。為了預防 SDRAM 頻率超過特定的頻率，SDRAM 必要時必須設定為 1/2 頻率。例如：從 100/100 變成 133/66，在頻率改變之前，SDRAM 匯流排必須設定為 1/2。而從 133/66 改變為 100/100，

則在完成頻率改變程序之後，SDRAM 必須設定為 1:1。相關細節請參考第 6 章「記憶體控制器」。

2. 關閉 LCD 控制器，或者將 LCD 控制器的組態設定成可以避免當應用處理器被中斷時，LCD 時脈會影響其資料。
3. 設定週邊單元的組態，讓週邊可以承受最多將有 500 μ s 的時間無法處理 DMA 的服務。如果週邊單元無法承受在 500 μ s 的時間無法得到 DMA 的服務時，則此週邊單元必須被關閉。
4. 在頻率改變程序期間所產生的中斷在頻率改變程序離開程序時，若週邊單元無法等待 500 μ s 的延遲時，此週邊單元必須被關閉。
5. 設定 CCCR(請參考 5.6.1 節「核心時脈組態暫存器」)來對應所需的頻率。

5.4.7.2 引發頻率改變程序

為了引發頻率改變程序，軟體必須設定 CCLKCFG 內的 FCS(請參考 5.7.1 節)。當軟體設定 FCS，也可以同時設定或清除 CCLKCFG 內其它的位元。若軟體在設定 FCS 的同時也對加速位元(Turbo bit)做設定，則當離開頻率改變程序後，CPU 才會進入加速模式(Turbo mode)。

在軟體設定 FCS 之後：

1. CPU 時脈將停止，而對 CPU 的中斷也會被控管。
2. 記憶體控制器會完成其緩衝區內未完成的交易，這些交易是來自 CPU 所產生的。其它來自 LCD 或 DMA 控制器所產生的新交易將會被忽略。
3. 記憶體控制器會將 SDRAM 設定為自我刷新模式(self-refresh mode)。

注意：為了確保 SDRAM 自我刷新時間在記憶體控制器內被正確地設定，必須將其值設定成可以符合核心目前的工作頻率與未來工作頻率兩者之間取最慢的頻率。

5.4.7.3 頻率改變程序期間之行為

在頻率改變程序時，處理器的 PLL 時脈產生器正在進行頻率鎖定的動作，而且不能被使用，這表示中斷無法被處理。在處理器的 PLL 頻率鎖定之後，便可以處理在頻率改變程序期間所產生的中斷。在頻率改變程序期間，95.85MHz 與 147.46MHz 的 PLL 時脈產生器仍然會繼續提供時脈輸出，除了記憶體控制器、LCD 控制器與 DMA 之外的週邊也可以繼續正常地操作，這些週邊可以適應沒有 DMA 或中斷請求的能力。直到頻率改變程序完成之前，DMA 或中斷請求是不會被認可的。

不正確資料放棄(Imprecise Data Abort)也不會被認可，若 nVDD_FAULT 或 nBATT_FAULT 被觸發，此觸發也會被忽略，直到頻率改變程序完成且離開為止。這表示處理器在沒有完成頻率改變程序之前是不會進入睡眠模式。

5.4.7.4 完成頻率改變程序

當任何重置被觸發時就會離開頻率改變程序。在硬體與看門狗重置發生時，進入與離開重置的優先權將高於頻率改變程序，而 PLL 會繼續其重置。在 GPIO 重置時，重置會被延遲，等待 PLL 重新調整時脈，使頻率設定成頻率改變程序所需的頻率時，重置才會繼續。

在頻率改變程序期間，如果看門狗計時器有致能，先設定看門狗符合暫存器(Watchdog Match Register)，讓頻率改變程序在看門狗重置觸發之前完成。

若在頻率改變程序期間觸發硬體或看門狗重置，DRAM 的內容會消失，因為所有的狀態，包含記憶體控制器組態與前一個頻率改變程序的資訊都會被重置。若在頻率改變程序期間觸發 GPIO 重置，若 SDRAM 不在自我刷新模式(self-refresh mode) 且離開程序超過刷新間隔(refresh interval)時，在離開 GPIO 重置程序後，SDRAM 的內容會消失。

正常情況下，頻率改變程序的離開程序如下：

1. 處理器的 PLL 時脈產生器以 CCCR 內的值來重新設定，並且開始調整時脈至鎖定的值。

注意：若改變前後的頻率相同，此步驟依然會執行。

2. 處理器時脈之內部 PLL 時脈產生器等待新的時脈達到穩定。相關細節請參考 *Intel® PXA250 and PXA210 Application Processors Electrical, Mechanical, and Thermal Specification*。
3. CPU 時脈重新啟動，而 CPU 會依照加速位元(TURBO bit)內所設定的狀態繼續操作(執行模式或加速模式)。傳至 CPU 的中斷不會再被阻隔。
4. FCS 位元不會自動清除。為了避免意外地返回頻率改變程序，軟體不可以立即清除 FCS 位元。此位元必須在下一次有需要的暫存器寫入時清除。
5. 數值可以被寫入至 CCCR，但它們會被忽略，一直到下一次頻率改變程序重新開始。
6. SDRAM 必須離開自行刷新模式而進入自己的閒置模式。關於 SDRAM 介面組態的相關細節請參考第 6 章「記憶體控制器」。

5.4.8 睡眠模式

睡眠模式提供低的電源消耗，以遺失大部分的處理器內部狀態做為代價。在睡眠模式時，應用處理器會遵循有順序的關機程序，並且移除核心的電源。電源管理員將會監視喚醒事件，而且在接收喚醒事件之後，會重新建立電源，並且遵循重置程序。在睡眠模式期間，RTC 與電源管理員會繼續運作。在整個睡眠模式期間，腳位的狀態從頭到尾皆可被控制維持在一定的狀態，而外部 SDRAM 的內容會被保留，因為它處於自行刷新模式。

因為睡眠模式開始時，除了 RTC 之外的所有應用處理器的活動都會停止，所以必須將週邊關閉以讓其有順序地關機。當離開睡眠模式時，處理器的狀態會重置，並以開機模式(boot-up mode)繼續運作。

5.4.8.1 睡眠模式之外部電壓調節器(External Voltage Regulator)需求

為了以最簡單的方式來執行睡眠模式，必須具有下列特徵的處理器內部元件，才能得到外部電壓調節器的電源供應：

- 會致能主要電壓供應輸出的電源致能腳位要與 VCC 與 PLL_VCC 有以下的關聯。此腳位必須連接至應用處理器的 PWR_EN 腳位。當 PWR_EN 被驅動為低電位時，調節器的輸出也應該驅動為低電位。當 PWR_EN 腳位為低電位狀態時，若 VCC 和 PLL_VCC 腳位為浮接，或者他們被驅動為正常的電壓值時，此時應用處理器仍然會正常運作，但電源消耗會較高。
- 較高電壓輸出會連接至 VCCQ 與 VCCN。當 PWR_EN 腳位被觸發時，這些調節器輸出會持續驅動而且不會改變。

5.4.8.2 睡眠模式之準備

在睡眠模式開始之前，軟體必須遵循以下的步驟。

1. 必須設定記憶體控制器組態來確定在睡眠模式期間，SDRAM 的內容可以被保持。詳細內容請參考第 6 章「記憶體控制器」。
2. 如果週邊有平順的關機需求時，則在睡眠模式被觸發之前就必須先關閉週邊。包含監視 DMA 對週邊的傳送或接收，或是監視 DMA 對記憶體的傳送或接收，以確定它們會完成。其它的週邊不需要關閉，因為它們在睡眠模式期間會保持在重置狀態。
3. 為了適當地進入與離開睡眠模式，下列的電源管理暫存器需要設定：
 - PM GPIO 睡眠狀態暫存器(PM GPIO Sleep State Register)(PGSR0、PGSR1、PGSR2)。當應用處理器準備喚醒時，為了避免產生匯流排競爭的情況，在睡眠模式期間，不可以將晶片選擇設定為 0。GPIO 的腳位可

以從外部提升(pull up)或下降(pull down)，或是改變設定為輸出，並且以不會觸發的電壓值來驅動。

- PM 通用組態暫存器浮接位元[FS/FP](PM General Configuration Register Float bits) 在系統裡必須適當地設定。通用組態暫存器浮接位元在喚醒時必須被清除。當應用處理器準備喚醒時，為了避免產生匯流排競爭的情況，在睡眠模式期間，不可以將晶片選擇設定為 0。
- 4. 必須設定下列的電源管理暫存器，用來偵測喚醒來源與振盪器的活動：
 - PM GPIO 睡眠狀態暫存器(PM GPIO Sleep State Register)(PGSR0、PGSR1 與 PGSR2)
 - PM 喚醒致能暫存器(PM Wake-up Enable Register - PWER)
 - PM GPIO 負緣偵測致能(PM GPIO Falling-edge Detect Enable Register - PFER)與 PM GPIO 正緣偵測致能暫存器(PM GPIO Rasing-edge Detect Enable Register - PRER)
 - 電源管理組態暫存器(Power Manager Configuration Register - PCFR)內之 OPDE 位元
 - PMCR 內之 IDAE 位元
- 5. 在設定 IDAE 位元之前，軟體必須確定當不正確資料放棄(Imprecise Data Abort)發生時會開始進入睡眠模式。當 nVDD_FAULT 或 nBATT_FAULT 觸發時，會產生一個放棄動作。這指出應用處理器正處於失去主電源供應的危險之中。

5.4.8.3 進入睡眠模式

軟體利用 PWRMODE 暫存器來進入睡眠模式(請參考 5.7.2 節)。

若外部電壓調節器(EVR)失效，或是主電池電源不足或消失時，則某些系統必須快速地進入睡眠模式。當 nBATT_FAULT 或 nVDD_FAULT 被觸發時，系統需要立即關機。

為了要讓 nVDD_FAULT 或 nBATT_FAULT 的觸發能引起一個不正確資料放棄(Imprecise data abort)，就必須設定 PMCR 內不正確資料放棄致能(IDAE)位元。設定 PMCR 內的 IDAE 位元會導致軟體執行資料放棄處理常式(data abort handler routine)，執行資料放棄處理常式也是進入睡眠模式的一部份。若 IDAE 位元被清除，應用處理器會立即進入睡眠模式，而不會執行放棄處理常式。

在進入睡眠模式期間，無法利用 PSSR[VFS]與 PSSR[BFS]來確定所產生的錯誤類型(VDD 錯誤或電池錯誤)。若 nVDD_FAULT 或 nBATT_FAULT 其中一個訊號被觸發，或兩個同時被觸發(PMCR 的 IDAE 位元被設定)，則軟體資料放棄

處理常式(software data abort handler)會被呼叫。由於只有一個共用的資料放棄處理常式，所以軟體第一個要務就是要確認出是 nVDD_FAULT 或是 nBATT_FAULT 觸發事件造成不正常資料放棄(讀取協力處理器 7、暫存器 4、位元 5(PSFS))。若 PSFS 位元被清除，則既不是 nVDD_FAULT 也不是 nBATT_FAULT 的觸發，所以資料放棄處理常式會是因為其它的原因被呼叫。若 PSFS 位元被設定，表示不是 nVDD_FAULT 就是 nBATT_FAULT 的觸發，但是可能無法辨別出是上述兩者的哪一個原因被觸發。不管是 nVDD_FAULT 或是 nBATT_FAULT 被觸發時，軟體應該儘快地依照下列步驟關閉系統，並且進入睡眠模式。

注意：在放棄處理常式內所使用的所有位址(包含資料與指令)應該為常駐的，且這些位址在記憶體分頁表內為可存取的，例如：系統軟體開發者應該確定當執行放棄處理常式時，不會有其他的放棄會發生才對。應用處理器並不支援遞迴的(巢狀的)放棄。在 nRESET_OUT 被觸發之前，系統不可以觸發 nBATT_FAULT 或 nVDD_FAULT 訊號超過一次。系統軟體也沒有辦法回復至 nBATT_FAULT 或 nVDD_FAULT 之前的正常執行。當放棄模式(abort mode)執行時，若發生電池或 VDD 錯誤，則會發生重新進入放棄處理常式的情形。這種遞迴放棄的情形是用軟體的方式偵測出來的，透過讀取保留程式狀態暫存器(Saved Program Status Register — SPSR)，來查看暫存器先前的內容是否正在執行放棄模式。

為了進入睡眠模式，軟體必須完成以下程序步驟：

1. 軟體使用外部記憶體與電源管理便條暫存器(Power Manager Scratch Pad Register — PSPR)來保留重要的狀態訊息。
2. 軟體在 PWRMODE[M]內設定睡眠模式。中斷會立即放棄睡眠模式，並且繼續正常的運算。
3. CPU 會等待管線內所有的指令都完成。
4. 記憶體控制器會先處理完其緩衝區內的未完成交易，與來自 CPU 的未完成交易。之後來自於 LCD 或 DMA 控制器的新交易都將被忽略。
5. 記憶體控制器將 SDRAM 置於自行刷新模式。
6. 電源管理員將 GPIO 的輸出腳位切換至相對應的睡眠狀態。各腳位的睡眠狀態是透過載入電源管理 GPIO 睡眠狀態暫存器(PGSR0、PGSR1、PGSR2)來設定。為了避免應用處理器喚醒時對於匯流排會產生競爭的現象，必須要確定相關晶片的晶片選擇(CS)在睡眠模式期間沒有被設定為 0。
7. CPU 時脈停止，並且移除核心的電源供應。
8. 移除 PWR_EN 的觸發。

當電源管理員取得來自記憶體控制器的指示之後，表示其已經處理完未完成的交易，並且將 SDRAM 設定為自行刷新模式，在 GPIO 栓鎖住 PGSR 值之前，將會有 8 個核心時脈週期的等待，在資料栓鎖之後的 4 個核心時脈週期，

nRESET_OUT 將被設定為低電位。

在某些系統內，不正確資料放棄(Imprecise Data Abort)所造成的延遲時間，會比系統使用一個已經失去作用的電源供應器內的殘存電力所能提供維持運作的時間還長。這種情況一般只會發生在當應用處理器在睡眠模式開始之前，正處於要離開電源模式或是離開頻率改變程序時才會發生。頻率改變程序就是這種改變電源模式的一個例子。在這些電源模式與頻率改變程序發生時，是不可設定 IDAE 位元的。如此做將允許應用處理器可以立即進入睡眠模式，但因此會失去任何在處理器內的重要狀態。

若 IDAE 位元沒有被設定，而當 nVDD_FAULT 或 nBATT_FAULT 腳位被觸發時，則睡眠程序會從上述的第 4 步驟開始動作。

5.4.8.4 睡眠模式之行為

在睡眠模式裡，除了 RTC 之外的所有處理器與週邊時脈都會被關閉。除了有效的喚醒信號、重置信號，以及 nBATT_FAULT 信號之外，應用處理器不會辨識任何的中斷或是任何外部腳位狀態的改變。

在睡眠模式時，若 nBATT_FAULT 信號被觸發，則 GPIO[1:0]會被設定為唯一的有效喚醒信號。

電源管理員會查看喚醒事件為何，是因為 CPU 在睡眠模式開始前所設定的，還是因為電源管理員依據所偵測到的錯誤狀態來設定的。為了偵測 GPIO 腳位上的正緣或負緣，正緣或負緣必須至少維持大於 1 個完整的時脈週期時間(32.768kHz)。電源管理員必須花費 3 個時脈週期(32.768kHz)來確認 GPIO 的正緣或負緣，以便開始喚醒程序。

關於在睡眠模式重置與其它重置期間，PXA250 應用處理器的腳位狀態，請參考表 4-7「PXA250 應用處理器腳位與訊號說明」。關於在睡眠模式重置與其它重置期間，PXA210 應用處理器的腳位狀態，請參考表 4-9「PXA210 應用處理器腳位與訊號說明」。

5.4.8.5 離開睡眠模式

當硬體重置被觸發時，會離開睡眠模式。硬體重置的進入與離開程序的優先

權高於睡眠模式。

注意：若硬體重置在睡眠模式期間被觸發，DRAM 的內容會消失，因為所有的狀態都會被重置，包括先前和睡眠模式有關的記憶體控制器組態與資料。

一般情況下，依照以下的程序離開睡眠模式。nBATT_FAULT 腳位在任何時間被觸發時，應用處理器都會回到睡眠模式。在外部電源供應穩定之計時器終止之前，nVDD_FAULT 腳位會被忽略。

1. 來自致能的 GPIO 或 RTC 的預先設定的喚醒事件發生時。若 nBATT_FAULT 腳位被觸發，則喚醒來源會被忽略。
2. PWR_EN 訊號被觸發，電源管理器會等待外部電源供應器達到穩定狀態。若 nVDD_FAULT 在外部電源供應計時器終止之後被觸發，應用處理器會回到睡眠模式。
3. 睡眠模式開始時若 PCFR[OPDE]與 OSCC[OON]被設定時，3.6864MHz 振盪器會被致能，並且達到穩定狀態。若 3.6864MHz 振盪器之前已經達到穩定狀態，則此步驟略過。
4. 處理器的 PLL 時脈產生器依照 CCCR 裡的值被重新程式化，並且達到穩定狀態。
5. 在 PWRMODE[M]內的睡眠模式組態被清除。
6. 應用處理器的內部重置的觸發被移除，CPU 開始正常開機程序。當正常開機程序開始時，除了 RTC、部分的時脈、電源管理器和記憶體控制器之外，應用處理器所有的單元都會回到它們預先定義的重置設定。
7. nRESET_OUT 腳位的觸發被移除。這代表應用處理器要對重置向量(reset vector)進行抓取的動作。
8. 存取 GPIO 之前，請先清除 PSSR[PH]，包括與 GPIO 腳位做多工處理的晶片選擇功能。
9. 若 PCFR[FS]或 PCFR[FP]任何一個在觸發睡眠模式之前被設定，則請清除。
10. SDRAM 必須從自動刷新模式轉變到閒置狀態。設定 SDRAM 介面的細節請參考第 6 章「記憶體控制器」。
11. 軟體必須透過檢查 RCSR 來確定造成重新開機的原因、透過檢查電源管理睡眠狀態暫存器(PSSR)來確定觸發睡眠模式的原因。
12. 在睡眠模式期間，若 PSPR 被用來保留任何重要的狀態，則軟體現在可以回復其資料。

在睡眠模式離開程序期間，若 nVDD_FAULT 或 nBATT_FAULT 腳位被觸發，系統會依以下程序再次進入睡眠模式。

1. 不管 IDAE 位元的狀態：
 - 清除所有 GPIO 邊緣偵測和 RTC 警示中斷。

- 電源管理喚醒來源暫存器(PWER、PRER 和 PFER)將被載入它們的喚醒錯誤狀態 0x0000 0003。這樣將限制 GPIO[0]和 GPIO[1]腳位上的正緣或負緣的喚醒來源。當電池電量不足或電源供應器供應不穩定時，設定喚醒錯誤狀態可預防假的事件，避免不希望發生的喚醒。
2. 關閉 PLL 時脈產生器。
 3. 若 PCFR 內的 OPDE 位元和 OSCC 內的 OON 位元被設定，則關閉 3.6864MHz 振盪器。若振盪器已關閉，則在睡眠模式時會消耗較少的電源。若振盪器被致能，則離開睡眠模式時會比較快。
 4. 核心及大部分的週邊模組將會產生內部重置，並觸發 nRESET_OUT 腳位。
 5. PWR_EN 腳位的觸發被移除，系統會已關閉 VCC 和 VCC_PLL 電源供應器做為 PWR_EN 腳位觸發移除的回應。為了使電源消耗達到最少，當 PWR_EN 為低電位驅動時，調節器(regulator)的輸出也必須為低電位驅動。若 PWR_EN 腳位為低電位驅動時，VCC 和 VCC_PLL 為浮接或是依照其正常值驅動時，則應用處理器會正常運作，並且電源消耗的情形會比正常情況稍微高一點。

5.4.9 電源模式概要

表 5-4 說明進入電源模式時發生的動作。表 5-5 說明電源模式離開時發生的動作。在表中的空格代表在電源模式時會跳過該步驟。表 5-6 說明在每個電源模式下電源供應的預期行為。

表 5-4 進入電源模式程序表

| 步驟 | 動作描述 | 加速(Turbo) | 執行(Run from Turbo) | 閒置(Idle) | 頻率改變(Freq Change) | 睡眠(Sleep) | 錯誤(Fault ¹ Sleep) |
|----|--------------------------|-----------|--------------------|----------|-------------------|-----------|------------------------------|
| 1 | 軟體寫入 CP14 裡的一個位元。 | × | × | × | × | × | |
| 2 | CPU 一直等待到完成所有指令。 | × | × | × | × | × | |
| 3 | 喚醒來源被清除，且 GP[1:0]被限制。 | | | | | | × |
| 4 | 電源管理員將 GPIO 設定為睡眠狀態。 | | | | | × | × |
| 5 | 記憶體控制器完成所有未完成的交易。 | | | | × | × | × |
| 6 | 記憶體控制器將 SDRAM 設定為自行刷新模式。 | | | | × | × | × |
| 7 | 關閉 PLL。 | | | | × | × | × |

| | | | | | | | |
|----|---------------------------------------|--|--|--|--|---|---|
| 8 | 若 OPDE 和 OOK 位元被設定，則關閉 3.6864MHz 振盪器。 | | | | | × | × |
| 9 | 大部分的模組被內部重置。nRESET_OUT 觸發被移除。 | | | | | × | × |
| 10 | PWR_EN 觸發被移除。電源被切斷。 | | | | | × | × |
| 11 | 大部分的 I/O 腳位的電源被切斷。 | | | | | | |

1：若 IDAE 被清除，且 nBATT_FAULT 或 nVDD_FAULT 被觸發時，則開始進入錯誤睡眠模式(Fault Sleep Mode)。

表 5-5 離開電源模式程序表

| 步驟 | 動作描述 | 加速(Turbo) | 執行(Run from Turbo) | 閒置(Idle) | 頻率改變(Freq Change) | 睡眠(Sleep) | 錯誤(Fault ¹ Sleep) |
|----|---|-----------|--------------------|----------|-------------------|-----------|------------------------------|
| 1 | 喚醒來源或中斷被接收。 | | | × | | × | × |
| 2 | 恢復 I/O 腳位的電源。 | | | | | | |
| 3 | PWR_EN 被觸發。 | | | | | × | × |
| 4 | 啟動外部電源。 | | | | | × | × |
| 5 | 若 OPDE 和 OOK 被設定，則致能 3.6864MHz 振盪器。 | | | | | × | × |
| 6 | 若 OPDE 和 OOK 被設定，則等待 3.6864MHz 振盪器達到穩定。 | | | | | × | × |
| 7 | 以新的頻率致能 PLL。 | | | | × | × | × |
| 8 | 等待 PLL 達到穩定。 | | | | × | × | × |
| 9 | 等待內部達到穩定。 | | | | | × | × |
| 10 | 清除 CP14 位元。 | | | × | | × | |
| 11 | 移除 nRESET_OUT 的觸發。 | | | | | × | × |
| 12 | 重新啟動 CPU 時脈，並致能中斷。 | × | × | × | × | × | × |

1：若 IDAE 被清除，且 nBATT_FAULT 或 nVDD_FAULT 被觸發時，則開始進

入錯誤睡眠模式(Fault Sleep Mode)。

表 5-6 電源模式期間之電源與時脈的供應來源及狀態

| 模組 (Module) | 供應來源 (Supply Source) | | 電源模式(Power Mode) | | | | | | | | | | |
|--------------------------------|-------------------------|------------------------|------------------|----|-------------|----|--------------|-----|--------------------------|---------|---------------|-----|----|
| | | | 加速 (Turbo) | | 執行 (Run) | | 閒置 (Idle) | | 頻率改變 (Freq change) | | 睡眠 (Sleep) | | |
| | Pw | Ck | Pw | Ck | Pw | Ck | Pw | Ck | Pw | Ck | Pw | Ck | |
| CPU、 快取記憶體、 緩衝區 | VCC | Run/ Turbo (R/T) | On | T | On | R | On | Off | On | 改變 中 | Off | Off | |
| 記憶體控制 器 | | Mem | | On | | On | | On | | | | | On |
| LCD 控制器 | | | | | | | | | | | | | |
| DMA 控制 器 | | | | | | | | | | | | | |
| 通用週邊 (General Periphs.) | | PLL | | On | | On | | On | | On | | | On |
| OS 計時器 | | 3.686 MHz Osc | | | | | | | | | | | |
| 中斷 | | | | | | | | | | | | | |
| 即時時脈 | VCC/ Reg (V/R) | 32.768 kHz Osc | V | On | V | On | V | On | V | On | I | On | |
| GP[3:0]、PM pads、Osc pads | HV/ Batt (H/B) | 動態/ 靜態 (D/S) | H | D | H | D | H | D | H | D | H | S | |
| 通用 I/O | H | | | | | | | | | | | | |

註解：

T：Turbo 時脈

R：Run 時脈

V：模組關閉 VCC

I：模組關閉內部調節器(regulator)

H：模組關閉 VCCQ 或 VCCN

D：模組為動態或主動時脈

S：模組為靜態或其時脈被閘控

5.5 電源管理員暫存器

此章節說明控制電源管理員的 32 位元暫存器。

5.5.1 電源管理員之控制暫存器

當 nVDD_FAULT 或 nBATT_FAULT 腳位為低電位觸發時，PMCR 用來選擇進入睡眠模式的方法。當 IDAE 位元被設定時，不正確資料放棄的指示被送至 CPU。CPU 便會執行一個放棄常式。軟體必須確定放棄常式設定 PWRMODE 暫存器內的睡眠模式組態(請參考 5.7.2 節「PWRMODE 暫存器(CP14 暫存器 7)」)。當離開睡眠模式時，任何的重置都會清除 IDAE 位元。當有需要時，軟體也可能會清除 IDAE 位元。必須經由記憶體管理單元(MMU)的權限來保護 PMCR。

表 5-7 電源管理員控制暫存器

| 0x40F0 0000 | | | | | | | | | | Power Manager Control Register (PMCR) | | | | | | | | | | Power Manager | | | | | | | | | | | | |
|-------------|----------|----|----|----|----|----|----|----|----|---------------------------------------|----|----|----|----|----|----|------|----|----|---------------|----|----|---|---|---|---|---|---|---|---|---|---|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Reserved | | | | | | | | | | | | | | | | IDAE | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

| 位元 | 名稱 | 說明 |
|--------|------|--|
| [31:1] | — | 保留。 讀取未定義，寫入必須為 0。 |
| 0 | IDEA | 不正確資料放棄致能 0=當 nVDD_FAULT 或 nBATT_FAULT 被觸發時，允許立即進入睡眠模式。 1=當 nVDD_FAULT 或 nBATT_FAULT 被觸發時，為了讓軟體進入睡眠模式，強制將不正確資料放棄訊號送至 CPU(建議模式)。 當硬體重置、看門狗重置、GPIO 重置，或離開睡眠模式時，該位元會被清除。 |

5.5.2 電源管理員之通用組態暫存器

PCFR 內含用來設定應用處理器功能的位元。當 OPDE 位元被設定時，在睡眠模式期間將允許關閉 3.6864MHz 振盪器。硬體重置、看門狗重置、GPIO 重置時，OPDE 位元會被清除。在睡眠模式期間，Float PCMCIA(FP) 位元和 Float Static Memory(FS)位元將會控制 PCMCIA 控制腳位的狀態和靜態記憶體控制腳位的狀態。

表 5-8 PCFR 暫存器位元地圖和位元定義

| 0x40F0 001C | | | | | | | | | | Power Manager General Configuration Register (PCFR) | | | | | | | | | | Power Manager | | | | | | | | | | | | |
|-------------|----------|----|----|----|----|----|----|----|----|---|----|----|----|----|----|----|----|----|----|---------------|----|----|---|---|---|---|---|----|---|----|---|------|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Reserved | | | | | | | | | | | | | | | | | | | | | | | | | | | FS | | FP | | OPDE |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| 位元 | 名稱 | 說明 |
|--------|------|--|
| [31:3] | — | 保留。 讀取未定義，寫入必須為 0。 |
| 2 | FS | 睡眠模式期間，浮接靜態晶片(Float Static Chip)的選擇。 0=在睡眠模式，靜態晶片選擇(Static Chip Select)不為浮接。nCS[5:1]將依照適當的 PGSR 暫存器位元的狀態而驅動。nCS[1]、nWE 和 nOE 將驅動為高電位狀態。 1=在睡眠模式期間，靜態晶片選擇(Static Chip Select)為浮接。nCS[5:0]、nWE 和 nOE 腳位將受影響。 在硬體重置、看門狗重置和 GPIO 重置時會清除此位元。 |
| 1 | FP | 睡眠模式期間，Float PCMCIA 的控制。 0=在睡眠模式，PCMCIA 的腳位不為浮接。它們將依照適當的 PGSR 暫存器位元的狀態而驅動。 1=在睡眠模式，PCMCIA 的信號：nPOE、nPWE、nPIOW、nPIOR 和 nPCE[2:1]為浮接。nPSKTSEL 和 nPREG 腳位的狀態將反應位址信號以及位址匯流排的狀態。 在硬體重置、看門狗重置和 GPIO 重置時清除此位元。 |
| 0 | OPDE | 3.6864MHz 振盪器電源關閉致能。 若 32.768kHz 晶體被關閉是因為振盪器組態暫存器(Oscillator Configuration Register)內的 OON 位元為 0，則 OPDE 將被忽略，且 3.6864MHz 振盪器不會被關閉。 0=睡眠模式期間不會停止 3.6864MHz 振盪器。 1=睡眠模式期間停止 3.6864MHz 振盪器。 |

| | | |
|--|--|-----------------------------|
| | | 在硬體重置、看門狗重置和 GPIO 重置時清除此位元。 |
|--|--|-----------------------------|

5.5.3 電源管理員之喚醒致能暫存器

表 5-9 為電源管理員之喚醒致能暫存器(PWER)內的所有喚醒來源致能位元的位置。若 GPIO 用來當成睡眠之喚醒來源，則其腳位必須在 GPDR 內程式化為輸入，而 PRER 和 PFER 內相對應的位元兩者之一或全部都必須被設定。當 IDAE 位元為 0，且錯誤狀態由 nVDD_FAULT 或 nBATT_FAULT 的腳位上被偵測出來時，PWER 會被設定為 0x0000 0003，且只允許 GP[1:0]當作喚醒來源。當 IDAE 位元被設定時，nVDD_FAULT 或 nBATT_FAULT 的錯誤狀態不會影響喚醒來源。在硬體重置、看門狗重置、GPIO 重置時，PWER 也會設定為 0x0000 0003。

表 5-9 PWER 暫存器位元地圖與位元定義

| 0x40F0 000C | | | | | | | | | | Power Manager Wake-Up Enable Register (PWER) | | | | | | | | | | Power Manager | | | | | | | | | | | | |
|-------------|-------|----------|----|----|----|----|----|----|----|--|----|----|----|----|----|----|------|------|------|---------------|------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WERTC | Reserved | | | | | | | | | | | | | | | WE15 | WE14 | WE13 | WE12 | WE11 | WE10 | WE9 | WE8 | WE7 | WE6 | WE5 | WE4 | WE3 | WE2 | WE1 | WE0 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | |

| 位元 | 名稱 | 說明 |
|---------|-------|---|
| 31 | WERTC | RTC 睡眠模式喚醒致能。 0=RTC alarm 關閉之喚醒。 1=RTC alarm 致能之喚醒。 在硬體重置、看門狗重置、GPIO 重置時清除此位元。 |
| [30:16] | — | 保留。 讀取未定義，寫入必須為 0。 |
| [15:0] | WEx | 睡眠模式喚醒致能 0=GPx 邊緣偵測關閉之喚醒。 1=GPx 邊緣偵測致能之喚醒。 在硬體重置、看門狗重置、GPIO 重置時設定為 0x 0003。 |

5.5.4 電源管理員之正緣偵測致能暫存器

PRER 暫存器(Power Manager Rising-edge Detect Enable register)可以確認是否因為 GPIO 腳位(由 PWER 暫存器使喚醒功能致能)上的正緣，造成從睡眠模式中喚醒。當 IDAE 位元為 0 且在 nVDD_FALUT 或 nBATT_FAULT 腳位偵測出錯誤狀況時，PRER 會被設定為 0x0000 0003。這將會把 GP[1:0]腳位上的正緣當作喚醒來源。當 IDAE 位元被設定時，在 nVDD_FAULT 或 nBATT_FAULT 腳位的錯誤狀況不會影響喚醒來源。在硬體重置、看門狗重置和 GPIO 重置時，PRER 也會設定為 0x0000 0003。

表 5-10 PRER 暫存器位元地圖與位元定義

| 0x40F0 0010 | | | | | | | | Power Manager Rising-edge Detect Enable Register (PRER) | | | | | | | | | | | | | | | | Power Manager | | | | | | | | |
|-------------|----------|----|----|----|----|----|----|---|----|----|----|----|----|----|----|----|------|------|------|------|------|------|-----|---------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Reserved | | | | | | | | | | | | | | | | RE15 | RE14 | RE13 | RE12 | RE11 | RE10 | RE9 | RE8 | RE7 | RE6 | RE5 | RE4 | RE3 | RE2 | RE1 | RE0 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | |

| 位元 | 名稱 | 說明 |
|---------|-----|---|
| [31:16] | — | 保留。 讀取未定義，寫入必須為 0。 |
| [15:0] | REx | 睡眠模式正緣喚醒之致能 0=關閉喚醒之 GPx 正緣偵測。 1=致能喚醒之 GPx 正緣偵測。 在硬體重置、看門狗重置、GPIO 重置時設定為 0x 0003。 |

5.5.5 電源管理員之負緣偵測致能暫存器

PFER 暫存器(Power Manager Falling-edge Detect Enable register)可以確認是否因為 GPIO 腳位(由 PWER 暫存器使喚醒功能致能)上的負緣，造成從睡眠模式中喚醒。當 IDAE 位元為 0 且在 nVDD_FALUT 或 nBATT_FAULT 腳位偵測出錯誤狀況時，PRER 會被設定為 0x0000 0003。這將會把 GP[1:0]腳位上的負緣當作喚醒來源。當 IDAE 位元被設定時，在 nVDD_FAULT 或 nBATT_FAULT 腳位的錯誤狀況不會影響喚醒來源。在硬體重置、看門狗重置和 GPIO 重置時，PRER 也會設定為 0x0000 0003。

表 5-11 PFER 暫存器位元地圖與位元定義

| 0x40F0 0014 | | | | | | | | Power Manager Falling-edge Detect Enable Register (PFER) | | | | | | | | | | Power Manager | | | | | | | | | | | | | | |
|-------------|----------|----|----|----|----|----|----|--|----|----|----|----|----|----|----|----|------|---------------|------|------|------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Reserved | | | | | | | | | | | | | | | | FE15 | FE14 | FE13 | FE12 | FE11 | FE10 | FE9 | FE8 | FE7 | FE6 | FE5 | FE4 | FE3 | FE2 | FE1 | FE0 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | |

| 位元 | 名稱 | 說明 |
|---------|-----------------|---|
| [31:16] | — | 保留。 讀取未定義，寫入必須為 0。 |
| [15:0] | FE _x | 睡眠模式負緣喚醒之致能 0=關閉喚醒之 GP _x 負緣偵測。 1=致能喚醒之 GP _x 負緣偵測。 在硬體重置、看門狗重置、GPIO 重置時設定為 0x 0003。 |

5.5.6 電源管理員之 GPIO 邊緣偵測狀態暫存器

PEDR 暫存器(Power Manager Edge Detect Status register)可以指示出哪一個 GPIO 腳位(由 PWER、PRER、PFER 暫存器使喚醒功能致能)使應用處理器從睡眠模式中喚醒。PEDR 內的位元值只能依照 GPIO 腳位所接收的正緣或負緣來反應設定。若 PRER 被設定，PEDR 的位元值只會針對正緣來反應設定；若 PFER 被設定，PEDR 的位元值只會針對負緣來反應設定。寫入 1 至 PEDR 的位元可以將其位元重置為 0。在硬體重置、看門狗重置和 GPIO 重置時，PEDR 位元會被重置為 0。

表 5-12 PEDR 暫存器位元地圖與位元定義

| 0x40F0 0018 | | | | | | | | | | | | | | | | Power Manager Edge Detect Status Register (PEDR) | | | | | | | | | | | Power Manager | | | | | | | | | | |
|-------------|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--|------|------|------|------|------|------|-----|-----|-----|-----|---------------|-----|-----|-----|-----|-----|--|--|--|--|--|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | |
| | Reserved | | | | | | | | | | | | | | | | ED15 | ED14 | ED13 | ED12 | ED11 | ED10 | ED9 | ED8 | ED7 | ED6 | ED5 | ED4 | ED3 | ED2 | ED1 | ED0 | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |

| 位元 | 名稱 | 說明 |
|---------|----|-----------------------|
| [31:16] | — | 保留。 讀取未定義，寫入必須為 0。 |

| | | |
|--------|-----|--|
| [15:0] | EDx | 睡眠模式之邊緣偵測狀態 0=GPx 腳位上之喚醒不會被偵測。 1=GPx 腳位上之喚醒會被偵測。 在硬體重置、看門狗重置、GPIO 重置時會清除。藉由 寫入 1 來清除其所對應的內容。 |
|--------|-----|--|

5.5.7 電源管理員之睡眠狀態暫存器

- PSSR 包含以下的狀態旗標：
- 當 PWRMODE 暫存器內的睡眠模式組態被設定且睡眠模式開始時，SSS 旗標(Software Sleep Status flag)會被設定。(請參考 5.7.2 節「PWRMODE 暫存器(CP14 暫存器 7)」)
 - 當 nBATT_FAULT 腳位被觸發時，應用處理器被喚醒之後的任何時間，電池錯誤狀態(Battery Fault Status - BFS)位元會被設定(即使應用處理器已經處於睡眠模式中)。
 - 當 nVDD_FAULT 腳位被觸發使應用處理器進入睡眠模式，在應用處理器被喚醒之後，VDD 錯誤狀態(VDD Fault Status - VFS)位元會被設定。
 - 當睡眠模式開始時，週邊控制保留 (Peripheral Control Hold - PH) 位元會被設定，並且指示 GPIO 腳位在睡眠模式狀態所保留的數值。
 - 在硬體重置、GPIO 重置和看門狗重置與睡眠模式時，讀取關閉維持(Read Disable Hold — RDH)位元會被設定。RDH 位元指示出所有應用處理器的 GPIO 輸入路徑都被關閉。為了允許致能 GPIO 的輸入腳位，軟體必須寫入 1 至 RDH 位元將其重置。

藉由寫入 1 來清除狀態旗標，寫入 0 至狀態位元不會有任何效果。硬體重置、看門狗重置和 GPIO 重置時會清除或設定 PSSR。

表 5-13 PSSR 暫存器位元地圖與位元定義

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|----------|----|-----|----|----|----|----|----|----|--|----|----|----|----|----|----|----|----|----|---------------|----|----|---|----------|---|-----|---|-----|---|-----|---|---|
| 0x40F0 0004 | | | | | | | | | | Power Manager Sleep Status Register (PSSR) | | | | | | | | | | Power Manager | | | | | | | | | | | | |
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Reserved | | | | | | | | | | | | | | | | | | | RDH | | PH | | Reserved | | VFS | | BFS | | SSS | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |
| 位元 | 名稱 | | 說明 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| [31:6] | — | | 保留。 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | | |
|---|-----|---|
| | | 讀取未定義，寫入必須為 0。 |
| 5 | RDH | <p>讀取關閉維持(Read Disable Hold)。</p> <p>0=根據 GPIO 的組態來設定 GPIO 腳位。</p> <p>1=關閉當作輸入的所有 GPIO 腳位接收器。在週邊和 GPIO 介面被設定組態之後、以及被使用之前，必須經由處理器清除。</p> <p>由硬體重置、看門狗重置和 GPIO 重置與睡眠模式設定。寫入 1 來清除此位元。</p> |
| 4 | PH | <p>週邊控制維持(Peripheral Control Hold)。</p> <p>0=根據 GPIO 的組態來設定 GPIO 腳位。</p> <p>1=維持 GPIO 腳位於睡眠狀態。當睡眠模式開始時就會設定。在週邊介面被設定組態之後，以及被處理器使用之前，必須由處理器來清除。</p> <p>由硬體重置、看門狗重置和 GPIO 重置來清除此位元。寫入 1 來清除此位元。</p> |
| 3 | — | <p>保留。</p> <p>讀取未定義，寫入必須為 0。</p> |
| 2 | VFS | <p>VDD 錯誤狀態</p> <p>0=自從上一次 CPU 或重置清除此位元之後，nVDD_FAULT 腳位就沒有被觸發過。</p> <p>1=nVDD_FAULT 腳位在 Run 或 Idle 模式時被觸發，並且引發晶片進入睡眠模式；此位元只有在被喚醒之後被設定。</p> <p>當 nVDD_FAULT 在睡眠模式被觸發時，此位元不會被設定。</p> <p>由硬體重置、看門狗重置與 GPIO 重置來清除此位元。</p> |
| 1 | BFS | <p>電池錯誤狀態</p> <p>0=自從上一次 CPU 或重置清除此位元之後，nBATT_FAULT 腳位就沒有觸發過。</p> <p>1=nBATT_FAULT 腳位有被觸發；此位元只在喚醒之後才會被設定。</p> <p>當 nBATT_FAULT 在睡眠模式被觸發時，此位元可被設定。</p> <p>由硬體重置、看門狗重置與 GPIO 重置來清除此位元。</p> |
| 0 | SSS | <p>軟體睡眠狀態</p> <p>0=自從上一次 CPU 或重置清除 SSS 之後，軟體就沒有經由睡眠模式位元來進入睡眠模式。</p> <p>1=晶片有藉由設定睡眠模式位元而進入睡眠模式。</p> |

| | | |
|--|--|-----------------------------|
| | | 由硬體重置、看門狗重置與 GPIO 重置來清除此位元。 |
|--|--|-----------------------------|

5.5.8 電源管理員之便條(Scratch Pad)暫存器

電源管理員包含一個 32 位元的暫存器，此暫存器可用來儲存任何格式的處理器組態資料。PSPR 是一個維持暫存器(holding register)，在睡眠模式時會仍然會繼續供電，經由硬體、看門狗與 GPIO 重置來重置。在 Run 和加速模式期間，可以將任何數值寫入 PSPR。在離開睡眠模式之後仍可以讀取此數值，可以用來表示處理器在進入睡眠模式之前的組態。

表 5-14 PSPR 暫存器位元地圖與位元定義

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---------------|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| 0x40F0 0008 | | | | | | | | | | | | | | | | Power Manager Scratch Pad Register (PSPR) | | | | | | | | | | | | | | | | Power Manager | | | | | | | | | | | | | | | |
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | | | | | | | | | |
| | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | | | |

| | | |
|--------|----|---|
| 位元 | 名稱 | 說明 |
| [31:0] | SP | 便條(Scratch Pad) 在睡眠模式時，32 位元字組會被保留。 由硬體重置、看門狗重置與 GPIO 重置來清除此位元。 |

5.5.9 電源管理員之 GPIO 睡眠狀態暫存器

當應用處理器進入睡眠模式時，PM GPIO 睡眠狀態暫存器(PGSR0、PGSR1、PGSR2)允許軟體去選擇每一個 GPIO 腳位的輸出狀態。當透過軟體、nBATT_FAULT 或 nVDD_FAULT 的要求要進入睡眠模式時，PGSR 暫存器的內容會載入 GPIO 的輸出資料暫存器，軟體可以經由 GPSR 與 GPCR 暫存器控制輸出資料暫存器。只有已經設定組態為輸出的腳位會反應新的狀態。輸出資料暫存器的所有位元都會被載入。當應用處理器再次進入執行模式時，在軟體重置 PSSR 的位元之前，GPIO 腳位會維持在程式化的睡眠狀態。若腳位由輸入再次組態為輸出，暫存器的最後的內容會驅動給該腳位。

表 5-15 PGSR0 暫存器位元地圖與位元定義

| 0x40F0 0020 | | | | | | | | | | Power Manager GPIO Sleep State Register 0 (PGSR0) | | | | | | | | | | Power Manager | | | | | | | | | |
|-------------|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|---------------|--|--|--|--|--|--|--|--|--|
| Bit | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reset | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| 位元 | 名稱 | 說明 |
|--------|-----|--|
| [31:0] | SSx | 若程式化為輸出，GPx 的睡眠狀態 0=睡眠模式期間，腳位驅動為 0。 1=睡眠模式期間，腳位驅動為 1。 由硬體重置、看門狗重置和 GPIO 重置來清除此位元。 |

表 5-16 PGSR1 暫存器位元地圖與位元定義

| 0x40F0 0024 | | | | | | | | | | Power Manager GPIO Sleep State Register 1 (PGSR1) | | | | | | | | | | Power Manager | | | | | | | | | |
|-------------|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|---------------|--|--|--|--|--|--|--|--|--|
| Bit | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reset | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| 位元 | 名稱 | 說明 |
|--------|-----|--|
| [31:0] | SSx | 若程式化為輸出，GPx 的睡眠狀態 0=睡眠模式期間，腳位驅動為 0。 1=睡眠模式期間，腳位驅動為 1。 由硬體重置、看門狗重置和 GPIO 重置來清除此位元。 |

表 5-17 PGSR2 暫存器位元地圖與位元定義

| 0x40F0 0028 | | | | | | | | | | Power Manager GPIO Sleep State Register 2 (PGSR2) | | | | | | | | | | Power Manager | | | | | | | | | |
|-------------|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|---------------|--|--|--|--|--|--|--|--|--|
| Bit | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reset | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| 位元 | 名稱 | 說明 |
|---------|-----|--|
| [31:17] | — | 保留。 讀取未定義，寫入必須為 0。 |
| [16:0] | SSx | 若程式化為輸出，GPx 的睡眠狀態 0=睡眠模式期間，腳位驅動為 0。 1=睡眠模式期間，腳位驅動為 1。 由硬體重置、看門狗重置和 GPIO 重置來清除此位元。 |

5.5.10 重置控制狀態暫存器(Reset Controller Status Register)

CPU 使用 RCSR 來確定最近一次重置的原因。以下 4 個方式可重置應用處理器：

- 硬體重置(Hardware Reset)
- 看門狗重置(Watchdog Reset)
- 睡眠模式(Sleep Mode)
- GPIO 重置(GPIO Reset)

不同的模組在各種重置類型期間的行為細節，請參考表 4-5「內部暫存器狀態的各種重置之影響」。

每個 RCSR 的狀態位元可經由不同的重置來源來設定，藉由寫入 1 至此位元來做清除。看門狗重置、睡眠模式和 GPIO 重置的 RCSR 的狀態位元皆為硬體重置 0。表 5-18 說明 RCSR 的狀態位元。

表 5-18 RCSR 暫存器位元地圖與位元定義

| 0x40F0 0030 | | | | | | | | | | Reset Controller Status Register (RCSR) | | | | | | | | | | Power Manager | | | | | | | | | |
|-------------|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|---------------|--|-----|--|-----|--|-----|--|---|--|
| Bit | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | Reserved | | | | | | | | | | GPR | | SMR | | WDR | | HWR | | | |
| Reset | | | | | | | | | | | | | | | | | | | | 0 | | 0 | | 0 | | 0 | | 1 | |

| 位元 | 名稱 | 說明 |
|--------|-----|---|
| [31:4] | — | 保留。 讀取未定義，寫入必須為 0。 |
| 3 | GPR | GPIO 重置 0=自從上一次 CPU 或硬體重置清除此位元，就沒有發過生 GPIO 重置。 1=自從上一次 CPU 或硬體重置清除此位元，有發生過 GPIO 重置。 藉由硬體重置與設定為 1 來清除此位元。 |
| 2 | SMR | 睡眠模式 0=自從上一次 CPU 或硬體重置清除此位元，就沒有發生過睡眠模式。 1=自從上一次 CPU 或硬體重置清除此位元，有發生過睡 |

| | | |
|---|-----|---|
| | | 眠模式。 藉由硬體重置與設定為 1 來清除此位元。 |
| 1 | WDR | 看門狗重置 0=自從上一次 CPU 或硬體重置清除此位元，就沒有發生過看門狗重置。 1=自從上一次 CPU 或硬體重置清除此位元，有發生過看門狗重置。 藉由硬體重置與設定為 1 來清除此位元。 |
| 0 | HWR | 硬體重置 0=自從上一次 CPU 或硬體重置清除此位元，就沒有發生過硬體重置。 1=自從上一次 CPU 或硬體重置清除此位元，有發生過硬體重置。 藉由硬體重置與設定為 1 來清除此位元。 |

5.5.11 電源管理員之暫存器位置

表 5-19 為與電源管理員相關的暫存器，實體位置可以用來存取這些暫存器。

表 5-19 電源管理員暫存器位置

| 位址 | 名稱 | 說明 |
|-------------|-------|------------------------------|
| 0x40F0 0000 | PMCR | 電源管理員之控制暫存器 |
| 0x40F0 0004 | PSSR | 電源管理員之睡眠狀態暫存器 |
| 0x40F0 0008 | PSPR | 電源管理員之便條暫存器 |
| 0x40F0 000C | PWER | 電源管理員之喚醒致能暫存器 |
| 0x40F0 0010 | PRER | 電源管理員之 GPIO 正緣偵測致能暫存器 |
| 0x40F0 0014 | PFER | 電源管理員之 GPIO 負緣偵測致能暫存器 |
| 0x40F0 0018 | PEDR | 電源管理員之 GPIO 偵測狀態暫存器 |
| 0x40F0 001C | PCFR | 電源管理員之通用組態暫存器 |
| 0x40F0 0020 | PGSR0 | GP[31-0]電源管理員之 GPIO 睡眠狀態暫存器 |
| 0x40F0 0024 | PGSR1 | GP[63-32]電源管理員之 GPIO 睡眠狀態暫存器 |
| 0x40F0 0028 | PGSR2 | GP[84-64]電源管理員之 GPIO 睡眠狀態暫存器 |
| 0x40F0 0030 | RCSR | 重置控制之狀態暫存器 |

5.6 時脈管理員暫存器

- 時脈管理員包含 3 個暫存器：
- 核心時脈組態暫存器(Core Clock Configuration Register - CCCR)
 - 時脈致能暫存器(Clock Enable Register - CKEN)
 - 振盪器組態暫存器(Oscillator Configuration Register - OSCC)

5.6.1 核心時脈組態暫存器

CCCR 控制核心時脈頻率，核心、記憶體控制器、LCD 控制器以及 DMA 控制器等的頻率皆由此取得。晶體頻率至記憶體頻率之倍數(L)、執行模式頻率至記憶體頻率之倍數(M)，與執行模式頻率至加速模式頻率之倍數(N)都在此暫存器內被設定。時脈頻率如下所示：

記憶體頻率 = 3.6864MHz 晶體頻率 × 晶體頻率至記憶體頻率之倍數(L)
執行模式頻率 = 記憶體頻率 × 執行模式頻率至記憶體頻率之倍數(M)
加速模式頻率 = 執行模式頻率 × 執行模式頻率至加速模式頻率之倍數(N)

以外部記憶體或 LCD 的需求來選擇 L 值。也可以將 L 設為常數，當改變 M 與 N 時，允許頻率在 Run 與加速模式之間改變，而不會干擾記憶體的設定。以匯流排頻寬的需求與最小核心效能的需求為基礎來選擇 M 值。以最高核心效能的需求為基礎來選擇 N 值。

表 5-20 CCCR 暫存器位元地圖與位元定義

| 0x4130 0000 | | | | | | | | | | Core Clock Configuration Register (CCCR) | | | | | | | | | | Clocks Manager | | | | | | | | | | | | |
|-------------|----|----|----|----|----|----|----|----|----|--|----------|----|----|----|----|----|----|----|----|----------------|----|----|---|---|---|---|---|---|---|---|---|---|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | | | | | | | | | | | Reserved | | | | | | | | | | Z | | M | | L | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |

| 位元 | 名稱 | 說明 |
|---------|----|---|
| [31:10] | — | 保留。 讀取未定義，寫入必須為 0。 |
| [9:7] | N | 執行模式頻率至加速模式頻率之倍數 加速模式頻率 = 執行模式頻率 × N 000 – 保留 |

| | | |
|-------|---|---|
| | | 001 – 保留 010 – 倍數 = 1 011 – 倍數 = 1.5 100 – 倍數 = 2 101 – 保留 110 – 倍數 = 3 111 – 保留 在硬體與看門狗重置時設定為 010。 |
| [6:5] | M | 記憶體頻率至執行模式頻率之倍數 $\text{記憶體頻率} = \text{晶體頻率} \times L$ 00 – 保留 01 – 倍數 = 1(執行模式頻率等於記憶體頻率) 10 – 倍數 = 2(執行模式頻率為記憶體頻率之 2 倍) 11 – 保留 在硬體與看門狗重置時設定為 01。 |
| [4:0] | L | 晶體頻率至記憶體頻率之倍數 00000 – 保留 00001 – 倍數 = 27(來自 3.6864MHz 晶體之記憶體頻率為 99.53MHz) 00010 – 倍數 = 32(來自 3.6864MHz 晶體之記憶體頻率為 117.96MHz) 00011 – 倍數 = 36(來自 3.6864MHz 晶體之記憶體頻率為 132.71MHz) 00100 – 倍數 = 40(來自 3.6864MHz 晶體之記憶體頻率為 147.46MHz) 00101 – 倍數 = 45(來自 3.6864MHz 晶體之記憶體頻率為 165.89MHz) 00110 至 11111 – 保留 在硬體與看門狗重置時設定為 00001。 |

5.6.2 時脈致能暫存器

CKEN 暫存器可以致能或關閉大部分週邊單元的時脈。為了達到最低的電源消耗，任何單元的時脈若不使用，則必須寫入 0 至相對應的位元以關閉時脈。

表 5-21 CKEN 暫存器位元地圖與位元定義

| 0x4130 0004 | | | | | | | | | | | | | | | | Clock Enable Register (CKEN) | | | | | | | | | | Clocks Manager | | | | | | | | | |
|-------------|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------------------------|----------|--------|--------|--------|--------|----------|----------|-------|-------|----------------|-------|----------|-------|-------|-------|-------|--|--|--|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | Reserved | | | | | | | | | | | | | | | CKEN16 | Reserved | CKEN14 | CKEN13 | CKEN12 | CKEN11 | Reserved | Reserved | CKEN8 | CKEN7 | CKEN6 | CKEN5 | Reserved | CKEN3 | CKEN2 | CKEN1 | CKEN0 | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | | | |

| 位元 | 名稱 | 說明 |
|---------|--------|--|
| [31:17] | — | 保留。 讀取未定義，寫入必須為 0。 |
| 16 | CKEN16 | LCD 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 15 | — | 保留。 讀取未定義，寫入必須為 0。 |
| 14 | CKEN14 | I2C 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 13 | CKEN13 | FICP 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 12 | CKEN12 | MMC 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 11 | CKEN11 | USB 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 必須設定此位元以允許 GP7 選擇性功能 1 上的 48MHz 時脈可以輸出。 |
| [10:9] | — | 保留。 讀取未定義，寫入必須為 0。 |
| 8 | CKEN8 | I2S 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |

| | | |
|---|-------|--|
| 7 | CKEN7 | BTUART 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 6 | CKEN6 | FFUART 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 5 | CKEN5 | STUART 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 4 | — | 保留。 讀取未定義，寫入必須為 0。 |
| 3 | CKEN3 | SSP 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 2 | CKEN2 | AC97 單元時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 1 | CKEN1 | PWM1 時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |
| 0 | CKEN0 | PWM0 時脈致能 0=單元時脈關閉 1=單元時脈致能 由硬體與看門狗重置來設定。 |

5.6.3 振盪器組態暫存器

OSCC 控制著 32.768kHz 振盪器的組態。OSCC 包含 2 個位元，分別為只可設定(set-only)的 32.768kHz OON 位元與唯讀(read-only)的 32.768kHz OOK 位元。OON 位元可以致能外部 32.768kHz 振盪器，該致能動作只可以經由軟體來設定。

當振盪器被致能後，最多需要耗費 10 秒達到穩定狀態。當振盪器達到穩定狀態時，應用處理器會設定 OOK 位元。

當 OOK 位元被設定，由 32.768kHz 振盪器產生時脈來供應 RTC 和電源管理員。此外 3.6864MHz 振盪器也會使用。OPDE 位元則允許 3.6864MHz 振盪器在睡眠模式時被關閉，而若 OOK 位元被清除時，則會忽略此 OPDE 位元(視同已被清除)。OOK 只可藉由硬體重置來重置。

表 5-22 OSCC 暫存器位元地圖與位元定義

| 0x4130 0008 | | | | | | | | Oscillator Configuration Register (OSCC) | | | | | | | | | | | | | | | | Clocks Manager | | | | | | | | | | | |
|-------------|----------|----|----|----|----|----|----|--|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|----------------|---|---|---|---|-----|-----|---|---|--|--|--|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | Reserved | | | | | | | | | | | | | | | | | | | | | | | | | | | | OON | OOK | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |

| 位元 | 名稱 | 說明 |
|--------|-----|---|
| [31:2] | — | 保留。 讀取未定義，寫入必須為 0。 |
| 1 | OON | 32.768kHz OON(只可寫入一次的位元) 0=32.768kHz 振盪器被關閉。3.6864MHz 振盪器(除以 112)供應時脈給 RTC 與電源管理員。 1=32.768kHz 振盪器被致能。OON 一旦被寫入，除了硬體重置之外不可被清除。 藉由硬體重置來清除此位元。 |
| 0 | OOK | 32.768kHz OOK(唯讀位元) 0=32.768kHz 振盪器被關閉或處於不穩定狀態。 3.6864MHz 振盪器(除以 112) 供應時脈給 RTC 與電源管理員。 1=32.768kHz 振盪器被致能(OON=1)而且穩定。供應時脈給 RTC 與電源管理員。 藉由硬體重置來清除此位元。 |

5.6.4 時脈管理員暫存器位置

表 5-23 為與時脈管理員相關的暫存器，而實體位址則可以存取這些暫存器。

表 5-23 時脈管理員暫存器位置

| 位址 | 名稱 | 說明 |
|-------------|------|--|
| 0x4130 0000 | CCCR | 核心時脈組態暫存器 (Core Clock Configuration Register) |
| 0x4130 0004 | CKEN | 時脈致能暫存器 (Clock Enable Register) |
| 0x4130 0008 | OSCC | 振盪器組態暫存器 (Oscillator Configuration Register) |

5.7 協力處理器 14：時脈與電源管理

協力處理器 14 包含 2 個暫存器來控制電源模式與程序：

- CP14 暫存器 6 – CCLKCFG 暫存器
- CP14 暫存器 7 – PWRMODE 暫存器

表 5-24 協力處理器 14 時脈與電源管理總覽

| 功能 | Rd 內資料 | 指令 |
|------------|-----------------------------------|---------------------------|
| 讀取 CCLKCFG | — | MCR p14, 0, Rd, c6, c0, 0 |
| 進入加速模式 | Turbo = 1 | MCR p14, 0, Rd, c6, c0, 0 |
| 進入頻率改變程序 | FCS = 1 (在同一寫入時加速模式位元可以被設定或清除) | MCR p14, 0, Rd, c6, c0, 0 |
| 進入閒置模式 | M = 1 | MCR p14, 0, Rd, c7, c0, 0 |
| 進入睡眠模式 | M = 3 | MCR p14, 0, Rd, c7, c0, 0 |

5.7.1 CCLKCFG 暫存器(CP14 暫存器 6)

CCLKCFG 暫存器用來進入加速模式與頻率改變程序。軟體執行表 5-24 中合適的功能以進入相對應的模式與程序。在時脈與電源管理員開始初始化所需的模式或程序之前，所有核心初始化的記憶體請求都會被完成。

為了確定當進入頻率改變程序時加速位元不會被改變，軟體必須做一個 read-modify-write 的動作。

表 5-25 CCLKCFG 暫存器位元地圖與位元定義

| CP14 Register 6 | | | | | | | | | | | | | | | | CCLKCFG | | | | | | | | | | | | | | | | CP14 | | | | | | | | | | | | | | | |
|-----------------|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|-----|------|-------|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | | | | | | | | | |
| | Reserved | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | FCS | | TURBO | | | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | | | | |

| 位址 | 名稱 | 說明 |
|--------|-----|---|
| [31:2] | — | 保留。 讀取未定義，寫入必須為 0。 在硬體、看門狗和 GPIO 重置與離開睡眠模式時會清除此位元。 |
| 1 | FCS | 頻率改變程序 0=沒有進入頻率改變程序 1=進入頻率改變程序 在硬體、看門狗和 GPIO 重置與離開睡眠模式時會清除此位元。 |
| 0 | 加速 | 加速模式 0=沒有進入加速模式 1=進入加速模式 在硬體、看門狗和 GPIO 重置與離開睡眠模式時會清除此位元。 |

5.7.2 PWRMODE 暫存器(CP14 暫存器 7)

PWRMODE 用來進入閒置與睡眠模式。軟體藉由寫入 PWRMODE[M]來選擇相對應的模式。在時脈與電源管理員初始化相對應的模式之前，所有核心初始化的記憶體請求會被完成。

表 5-26 PWRMODE 暫存器位元地圖與位元定義

| CP14 Register 7 | | | | | | | | | | | | | | | | PWRMODE | | | | | | | | | | | | | | CP14 | | | | | | | | | | | | | |
|-----------------|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|----|----|----|----|----|----|---|---|---|---|---|---|---|------|---|---|--|--|--|--|--|--|--|--|--|--|--|
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | | | | | |
| | Reserved | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | M | | | | | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | |

| 位址 | 名稱 | 說明 |
|--------|----|-----------------------|
| [31:2] | — | 保留。 讀取未定義，寫入必須為 0。 |
| [1:0] | M | 低電源模式 00 – 執行／加速模式 |

| | | |
|--|--|---|
| | | 01 – 閒置模式 10 – 保留 11 – 睡眠模式 重置時設定為 00。 |
|--|--|---|

5.8 外部硬體的考量

時脈與電源管理員將控制重置時序的進入與離開、電壓供應和穩定。和應用處理器一起作用的硬體必須符合這些需求得以適當地操作。這個章節將說明這些需求。

5.8.1 電源供應程序

為了避免高電流供應的困境，應用處理器的電源供應必須以下列程序供電：

1. VCCQ
2. VCCN
3. VCC 與 PLL_VCC

注意：VCCN 不可在 VCCQ 之前供電。

5.8.2 通電開機之重置(Power-On-Reset)

在電源供應初始化並達到穩定之後的一段固定時間內，nRESET 與 nTRST 腳位必須維持低電位。這可以藉由外部通電開機之重置(Power-On-Reset)的裝置或其它電路來控制。

為了確保在通電開機期間內部 ESD 保護裝置不會啟動，必須觀察最小上升時間。相關細節請參考 *Intel® PXA250 and PXA210 Application Processors Electrical, Mechanical, and Thermal Specification*。

5.8.3 電源供應之連接

應用處理器需要 2 或 3 種外部供應電壓。VCCQ 需要 3.3V(+/-10%)、VCCN 需要 3.3V(+/-10%)或 2.5V(+15/-5%)，而 VCC 和 PLL_VCC 應該要連接在一起且

需要 0.85 ~ 1.3V。PLL_VCC 必須要與其它低電壓供應器被分離。依照可以取得的獨立調節器輸出和所需求的記憶體電壓，VCCQ 也最好從 VCCN 中分離出來。

5.8.4 從外部時脈來源來驅動晶體腳位

此章節提供使用指南。晶體振盪器腳位的電子規格請參考 *Intel® PXA250 and PXA210 Application Processors Electrical, Mechanical, and Thermal Specification*。

3.6864MHz 晶體必須連接到 PXTAL 和 PEXTAL 腳位。32.768kHz 晶體通常是連接到 TXTAL 和 TEXTAL 腳位。如此的連接組態提供整體最低的電源消耗，因為晶體的共振本質用來驅動晶體腳位，可以提供比利用外部來源更好的電源使用效率。一些應用程式有其它頻率相同的時脈來源，如此可以藉由外部來源用以驅動晶體腳位，以減少整體成本。

關於更多的振盪器電子規格請參考 *Intel® PXA250 and PXA210 Design Guide*。

注意：不需要使用外部電容。

5.8.5 晶體驅動腳位與晶體振盪器之間的雜訊耦合

二組晶體腳位在應用處理器上的位置相當接近。當晶體振盪器連接至該腳位，將產生低信號擺動(swing)與慢邊緣(slow edge)的現象，引發有限的雜訊耦合現象。若晶體振盪器的其中之一被獨立訊號來源所替換時(兩者之一的另一個沒有被替換)，雜訊耦合的現象可能會增加。為了限制此現象所引發的影響，要利用獨立的來源來驅動該腳位，藉此減少旋轉率(slew rate)。

問題：

1. 請說明時脈管理員之功能為何？
2. 請問時脈管理員如何決定內部週邊的頻率為何？
3. 請說明電源管理員之功能為何？
4. 請說明 Intel XScale 應用處理器有哪些重置方法？其特性與功能為何？
5. 請問 Intel XScale 應用處理器有哪些電源模式？這些模式有何特性？
6. 請問 Intel XScale 應用處理器有哪些主要的時脈來源？各供給哪些週邊時脈？
7. 32.768 KHz 振盪器提供 RTC 和電源管理員時脈。請問在沒有此震盪器時，

該如何提供 RTC 和電源管理員時脈?且該如何設定?

8. 請問在 Intel XScale 應用處理器的核心相位鎖定迴路(PLL)供給時脈給哪些模組? 其頻率是由哪些值來決定?
9. 請問系統的架構具有何種特性? 該設定哪些參數? 則可使各模組得到最高的頻率?
10. 請問如何引發硬體重置? 在硬體重置期間的行為為何? 如何離開硬體重置?
11. 請問如何引發看門狗重置? 在硬體重置期間的行為為何? 如何離開看門狗重置?
12. 請問如何引發 GPIO 重置? 在硬體重置期間的行為為何? 如何離開 GPIO 重置?
13. 請問何為加速模式? 如何進入加速模式? 在加速模式其間的行為為何? 如何離開加速模式?
14. 請問何為閒置模式? 如何進入閒置模式? 在閒置模式其間的行為為何? 如何離開閒置模式?
15. 請問何為頻率改變程序? 如何引發頻率改變程序? 在頻率改變程序其間的行為為何? 如何完成頻率改變程序?
16. 請問何為睡眠模式? 如何進入睡眠模式? 在睡眠模式其間的行為為何? 如何離開睡眠模式?