ARM®

Intelligent Energy Manager (IEM) Hardware Control System in the ARM1176JZF-S Development Chip

Application Note 172

Released on: November 2006

Intelligent Energy Manager (IEM) Hardware Control System in the ARM1176JZF-S Development Chip
Application Note 172

Release Information

Table 1 lists the changes to this document.

Table 1 Change history

Date	Issue	Confidentiality	Change
August 2006	A	Non-Confidential	Initial issue
November 2006	B	Non-Confidential	Updated for compatibility with InfoCenter

Proprietary Notice

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by ARM and the party that ARM delivered this document to.

Product Status

The information in this document is final, that is for a developed product.

Web Address

`http://www.arm.com`

1 Introduction

To understand the control system context, the development vehicle is presented briefly. The ARM1176 development chip used a TSMC 130nm G process, using ARM IP including ARM Artisan™ cell libraries, RAMs, flip-chip and specialty I/Os, and timing components (PLLs and DLLs).

This white paper documents the Intelligent Energy Manager (IEM) hardware control system implemented by ARM in the ARM1176JZF development chip. The development chip is a fairly simple agnostic SoC, implementing an application processor sub-system, comprising the ARM1176JZF core, Level 2 cache, and ETM11 trace. System components are a memory backplane comprising of static and dynamic memory controllers (Mobile DDR) and various peripherals to support OS booting and user interfaces. To assist customers in developing their own AMBA 3.0 AXI peripherals, AXI Master and Slave busses are brought off chip synchronously using a board-level de-skewed system clock. Figure 1 on page 4 shows a block diagram of the ARM1176JZF-S development chip.

For IEM to work efficiently there is a requirement that both clock frequency and voltage can be switched very quickly. Typically the IEM software requires the performance to be updated approximately every 50ms. However, certain OS events such as a context switches, interrupt and task creation or deletion can occur and cause new performance requests in a much shorter timeframe.

For the ARM1176JZF-S development chip, the National Semiconductor LP5550 PowerWise Energy Management Unit is used. This unit has extremely fast voltage switching times, and in typical conditions can switch voltage in less than 50µs.

For maximum flexibility two variable PLLs and one fixed PLL are used. The variable PLLs can be dynamically configured to provide a wide range of IEM frequencies, while the fixed PLL is always set at maximum CPU frequency. The fixed PLL therefore allows the IEM control system to switch very quickly to maximum performance thus enabling the CPU to service high priority events with maximum efficiency.

This document assumes that the reader has a thorough understanding of IEM, the ARM IEC, and APC1 SoC components.

The above techniques for implementing DVFS can be applied easily to the following ARM processor cores from the ARM1176 family, ARM11 MPcore and ARM Cortex family.

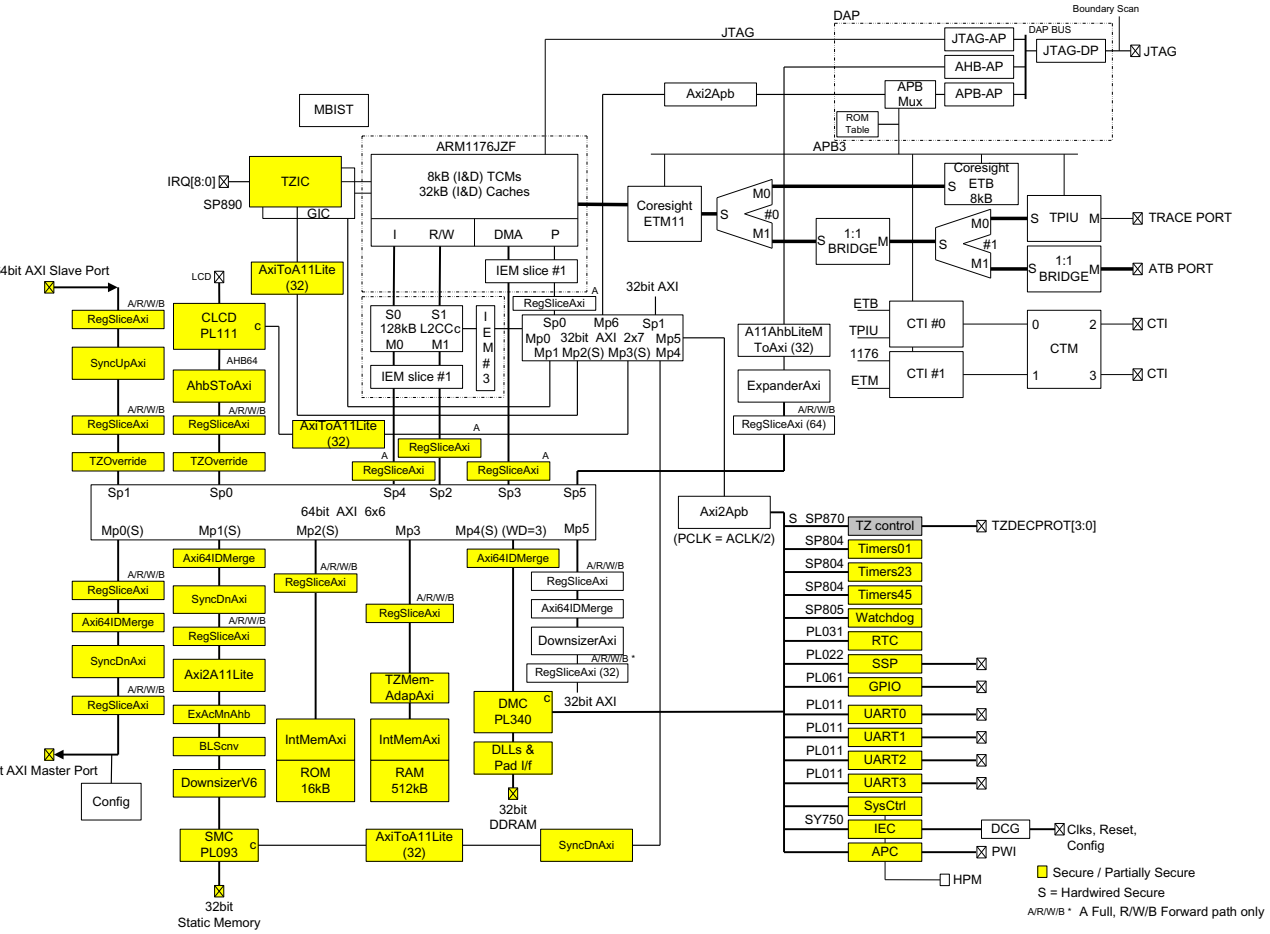

Figure 1 ARM1176JZF-S development chip block diagram

1.1 IP Revisions

The following ARM IP Revisions are used in the ARM1176JZF-S Development Chip.

Table 2 ARM IP revisions used in the ARM1176JZF-S development chip

IP Product	Revision
ARM1176JZF-S Processor Core	r0p0
Level 2 Cache Controller (L2CC)	r1p1
CoreSight ETM11 (Embedded Trace Macrocell)	r0p0
Advanced Power Controller (APC1) and Hardware Performance Monitor (HPM)	r0p0-00bet0
Intelligent Energy Controller	r0p0-00rel0 Hardware workaround for Errata. 2.

2 Example IEM Control System

Figure 2 shows a block diagram of the IEM control system as implemented in the ARM1176JZF-S development chip. The design supports both open-loop (DVS) and closed-loop (AVS) modes of IEM operation.

The complete control system comprises the Intelligent Energy Control (IEC) from ARM, Adaptive Power Controller (APC1), and Hardware Performance Monitor (HPM) from National Semiconductor (licensed from ARM), and a number of SoC-specific components.

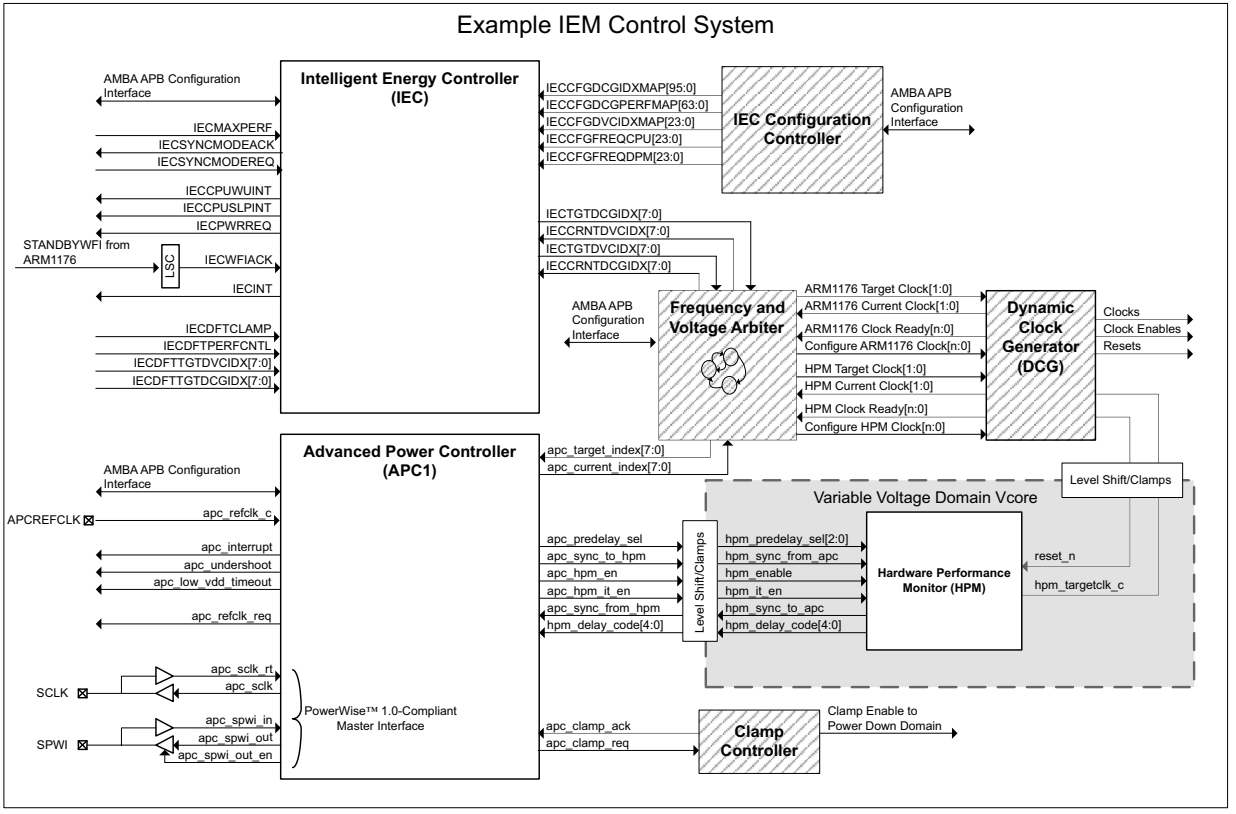

Figure 2 IEM Control System

2.1 System Overview

The IEC and APC1 use the same thermometer encoding scheme to transfer performance level requests and to indicate the current levels of both frequency and voltage. It is therefore possible to connect the IEC target voltage index **IECTGTDVCIDX**, and current voltage index **IECCRNTDVCIDX**, directly to the APC1 target and current voltage index control buses, **apc_target_index** and **apc_current_index**, respectively. In a similar manner, you can also connect the IEC target frequency index **IECTGTDCGIDX**, and current frequency index **IECCRNTDCGIDX** directly to a compatible Dynamic Clock Generator (DCG).

However, depending on the SoC-specific DCG design and supported power modes, it might be necessary to implement a custom frequency and voltage arbiter.

The main reason for this is that the IEC employs an ideal performance level switch model. This model expects that both voltage and frequency can be switched to new levels in zero time. This means that when the IEM software issues a new performance level request, the IEC immediately updates both the **IECTGTDCGIDX** and **IECTGTDVCIDX** outputs to reflect the new target levels for both frequency and voltage.

For a real system, in which both the frequency and voltage take a finite amount of time to switch, this might result in the voltage being reduced before the frequency, or the frequency being increased before the voltage. Both of these conditions can be very dangerous for the stability of the system. A frequency and voltage arbiter must therefore be used to protect against these conditions.

The ability to switch performance levels is very specific to each SoC design, and can vary depending on the DCG and Power Management Unit (PMU) used. By using the ideal switch model within the IEC, the end user has the maximum flexibility to develop an efficient IEM control system for each target application.

IEC Configuration Controller

The IEC provides a comprehensive configuration interface. As detailed in the IEC Technical Reference Manual ref[1], it is recommended that this configuration is fixed during RTL configuration of the design. However, depending on the deployment of the design it might be beneficial to make the IEC configuration software programmable. This can be particularly advantageous if the SoC is to be used for a number of applications with different performance level mappings and IEM modes of operation.

For this purpose, a simple AMBA APB slave can be created and used to set the IEC configuration interface.

Clamp Controller

When entering either shutdown or dormant mode, outputs from each powered-down domain must be held at safe levels. To ensure this is the case, the APC1 uses a simple clamp handshaking routine. When a power-down request has been made to the APC1, **apc_clamp_req** is asserted. The clamp controller must ensure that all domains to be powered down are safely clamped and then acknowledge this to the APC1 by asserting **apc_clamp_ack**. The APC1 can now make the power-down request to the LP5550 PMU using the PowerWise™ interface.

When returning from a power-down state after the power has been safely restored, **apc_clamp_req** is deasserted by the APC1. When this occurs, the clamp controller must disable the clamp circuitry and indicate this back to the APC1 by deasserting **apc_clamp_ack**.

2.2 Open-Loop Operating Mode DVFS

Open-loop mode is the simplest control method for implementing IEM. Each frequency index has a corresponding voltage index that is set by the configuration interface on the IEC. The frequency index is translated to a set frequency by the DCG, and the voltage index is translated within the APC1 to a programmed voltage level. This voltage is then requested using the PowerWise™ interface to an off-chip PMU such as the LP5550. For open-loop mode the DCG must be able to accept a target frequency from the frequency and voltage arbiter, and accurately report the currently selected frequency.

Important - When receiving new target frequency requests, it is imperative that the DCG does not stop the ARM1176 clock for long periods while switching, for example, while waiting for a PLL to lock to a new frequency. However, depending on the DCG design, it is probably unavoidable to drop a few clock cycles while resynchronizing to the new clock. In addition, it requires that no glitches are created during the clock switching process.

Figure 3 on page 9 shows an example frequency and voltage arbiter state machine for changing performance levels in open-loop mode.

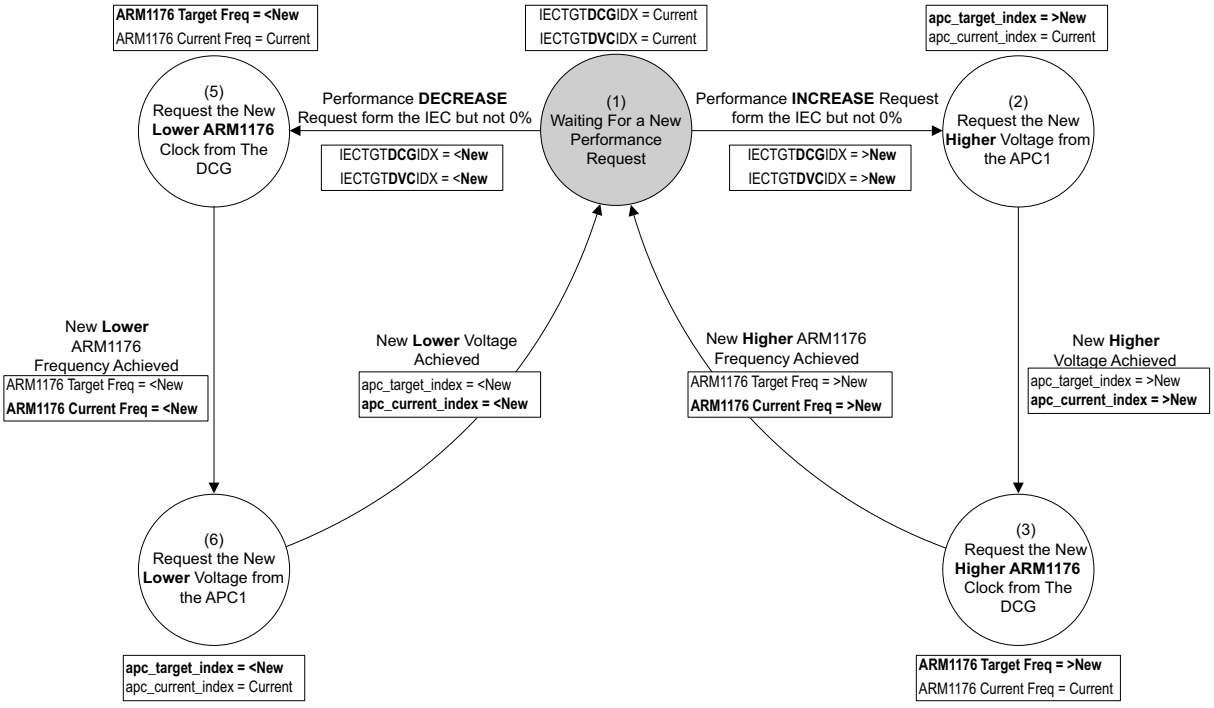

Figure 3 Example frequency and voltage arbiter state machine for open-loop mode

When a new target DCG and DVC index is requested from the IEC, the frequency and voltage arbiter must first assess whether the new performance request is an increase or decrease in performance. If it is an increase, the following steps must be taken by the frequency and voltage arbiter to ensure a safe performance transition.

- Software requests a performance increase. This results in a transition from state 1 to state 2 as Figure 2 on page 6.
- Because this is a performance increase, the voltage might not be sufficient to support the new target frequency. The new target index to the DCG is therefore blocked, and only the new target voltage index **apc_target_index** to the APC1 is made by the frequency and voltage arbiter.
- When the APC1 confirmed that the voltage has been successfully increased by updating **apc_current_index**, the state machine can safely move onto state 3.

- The new target frequency is then requested from the DCG by updating **ARM1176 Target Freq**. The frequency and voltage arbiter must also update **IECCRNTDVCIDX** to ensure that the IEM software can accurately track the current voltage.
- After the DCG has successfully switched frequencies, **ARM1176 Current Freq** is updated.
- The frequency and voltage arbiter must now update **IECCRNTDCGIDX**. This ensures that the Dynamic Performance Monitors within the IEC increment correctly, and that the IEM software can accurately monitor the amount of work performed within the system.

If a performance decrease is requested, a very similar flow is taken with the exception that the frequency and voltage arbiter must first ensure that the DCG has successfully switched to the lower frequency before the request is made to the APC1 to reduce the voltage.

The example shown here is the simplest form of open-loop control. Depending on specific system requirements and switching speeds between performance levels, it might be beneficial to allow new performance requests to override the initial target. This can be particularly useful when high priority software events or interrupts occur that require immediate maximum performance. For this example 0% power-down modes are not considered. For more information on these, refer to *Dormant and Shutdown Mode* on page 24.

2.3 Closed-Loop Mode DVFS (AVS)

For closed-loop mode, in addition to controlling the requested voltage to the APC1 and the clock to the ARM1176, the frequency and voltage arbiter and DCG must control the clock to the HPM.

Important - Although the HPM is delivered as soft IP, extra care must be taken during its implementation. To assist in this, detailed layout scripts are provided as part of the standard deliverables. Also prior to implementation, the HPM must be configured depending on the target process technology and operating frequency of the design. For more details on this, refer to the HPM Implementation Guide ref[2].

The clock to the HPM **hpm_targetclock_c** must be directly related to the target ARM1176 clock. For the ARM1176JZF-S development chip to suit the maximum CPU clock frequency and the HPM configuration implemented, the HPM clock is configured to always be the ARM1176JZF-S target clock divided by 4.

Unlike open-loop mode in which **apc_target_index** is directly related to a pre-set voltage within the APC1, for closed-loop mode the target apc index is used to indicate to the APC1 that a change in performance is imminent. Therefore a change to the HPM

hpm_targetclock_c clock is expected. To avoid a false voltage slack indication from the HPM when the APC1 detects an update to **apc_target_index**, it immediately disables the closed-loop voltage control. This false voltage slack indication can occur while updating the **hpm_targetclock_c** clock.

The current system voltage is held until the programmable **VDD_DELAY** timer within the APC1 expires. This timer must be set to the worst case time between updating **apc_target_index**, and when the **hpm_targetclock_c** clock becomes stable at the new target frequency. This is explained in more detail in *Example Closed-Loop Performance Switching* on page 12.

After the **VDD_DELAY** timer has expired, closed-loop control is enabled and the voltage is adjusted to suit the next target performance. When the target voltage has been achieved, the APC1 indicates this by updating **apc_current_index** with the initial target index requested on **apc_target_index**.

Because the way how the APC1 uses the **apc_target_index** to change voltage levels in closed-loop mode, although the actual value is arbitrary, it must be updated for each performance change.

Therefore, when using the APC1 in closed-loop mode, the frequency to voltage index mapping must be configured in 1-to-1 mode as explained in *Closed-Loop 1-to-1 Frequency to Voltage Mapping*. This is controlled by the **IECCFGDVCIDXMAP** configuration bus on the IEC.

Closed-Loop 1-to-1 Frequency to Voltage Mapping

Figure 4 shows an example of mappings between frequency and voltage index levels. This figure shows an IEM system with 7 frequency levels, but only 2 voltage levels. In this example, frequency index levels 1-4 all map to voltage index 1, and frequency index levels 5-7 all map to voltage index 7.

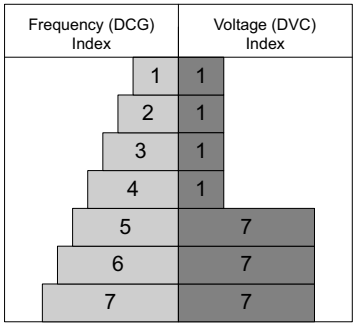

Figure 4 Open-loop mappings

For open-loop mode, this configuration is acceptable because the mappings between frequency and voltage are completely independent, and each voltage index directly relates to a voltage programmed within the APC1.

However, for closed-loop mode, the mappings between frequency and voltage index must be configured in 1-to-1 mapping mode. This means that for every unique frequency index level, there must be a corresponding unique voltage index level. Figure 2 4 shows an example of two frequency-to-voltage index mappings. The diagram on the left of the figure shows a system with 7 performance levels, and on the right is a system with 4 performance levels. Both of these configurations use 1-to-1 mapping, and are suitable for closed-loop mode of operation.

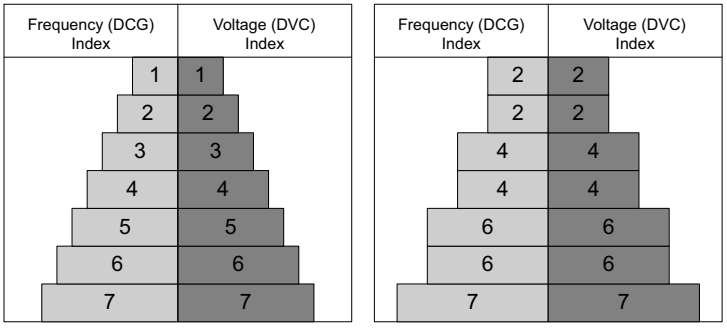

Figure 5 Figure 2 4 Closed-Loop 1-to-1 mappings

Example Closed-Loop Performance Switching

Figure 6 on page 13 shows an example frequency and voltage arbiter state machine for changing performance levels in closed-loop mode.

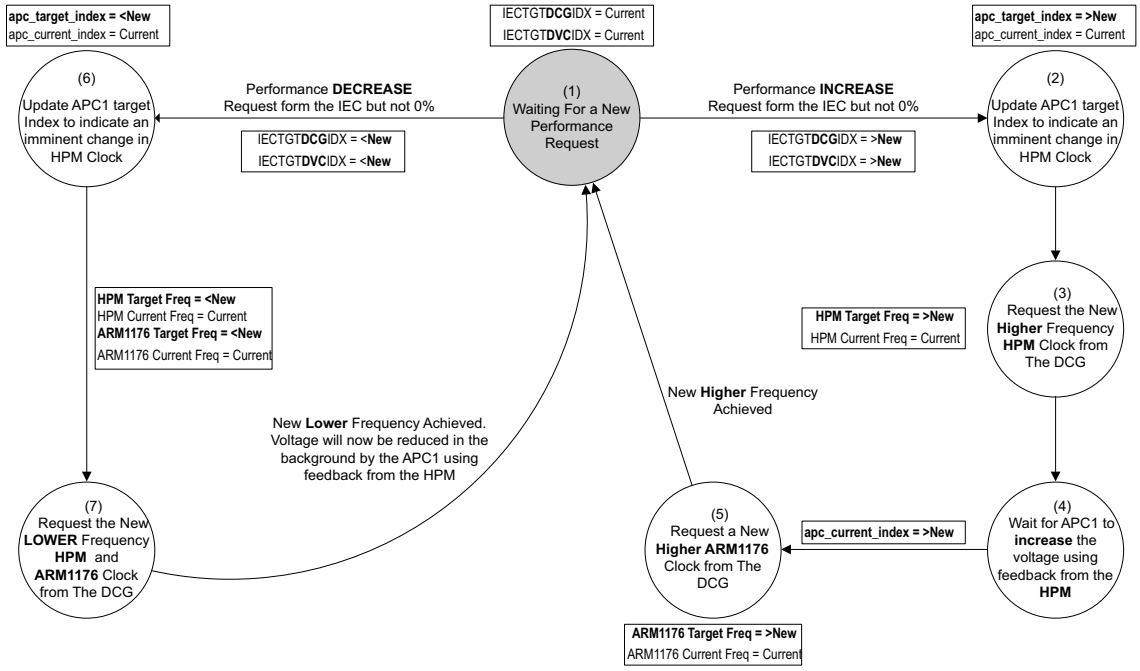

Figure 6 Example frequency and voltage arbiter state machine for closed-loop mode

APC1 indicates that the new voltage has been reached by looping back the **apc_target_index** to **apc_current_index**. For this reason when operating in closed-loop, a 1-to-1 mapping must be used between target DCG and DVC index. The DCG must have independent control of both the HPM and ARM1176 clocks.

When decreasing in performance, the ARM1176 and HPM clocks can be switched simultaneously providing that the HPM clock is not updated before **apc_target_index**, and the **VDD_DELAY** timer is greater than the time taken to switch both the HPM and ARM1176 clocks. If this is not the case, and the HPM clock is updated and evaluated by the APC1 before the ARM1176 clock has been reduced, the voltage can be reduced. Therefore, it is not sufficient for the current ARM1176 frequency level.

For closed-loop mode a Hardware Performance Monitor (HPM) is used. For the ARM1176JZF-S development chip, the HPM is provided by National Semiconductor. The HPM is a delay line based analog to digital converter. It provides gate delay information with respect to process, temperature, and voltage. The advantage of using the HPM in a closed-loop system is that the supplied voltage can be continually adjusted to ensure that the minimum voltage to support the current performance is always

applied. This reduces voltage slack in the system, and therefore consumes less energy. For full details of the APC1 and HPM, refer to the National Semiconductor document ref[2].

For the HPM to evaluate whether the current voltage is sufficient to support the required frequency, the HPM is provided with a sampling clock directly related to the target system clock. For the ARM1176JZF-S development chip, the HPM clock is the target clock divided by 4.

When switching to a higher frequency, it is suggested that the **Target_Performance_Index** is updated simultaneously to both the DCG and APC1. This can potentially cause a problem if the sampling clock to the HPM is not updated fast enough. In this situation the APC1 might incorrectly assume that the voltage is sufficient for the target frequency when in fact it is still evaluating the voltage for the current frequency. To prevent this, a timer **VDD_DELAY** is used in the APC1 to delay the time between the **Target_Performance_Index** changing, and HPM evaluation. However, this is not ideal because the delay timer is fixed, and must be set to the maximum possible time for the HPM clock to be updated. This might not be acceptable for our system, because the PLL lock time is unpredictable. Although under normal circumstances it should be less than 50µs, the worst case locking time can be a few milliseconds. This obviously has a considerable impact on the efficiency of the IEM control system performance in closed-loop mode. Figure 7 shows the timing diagram for this standard mode of operation for closed-loop performance switching.

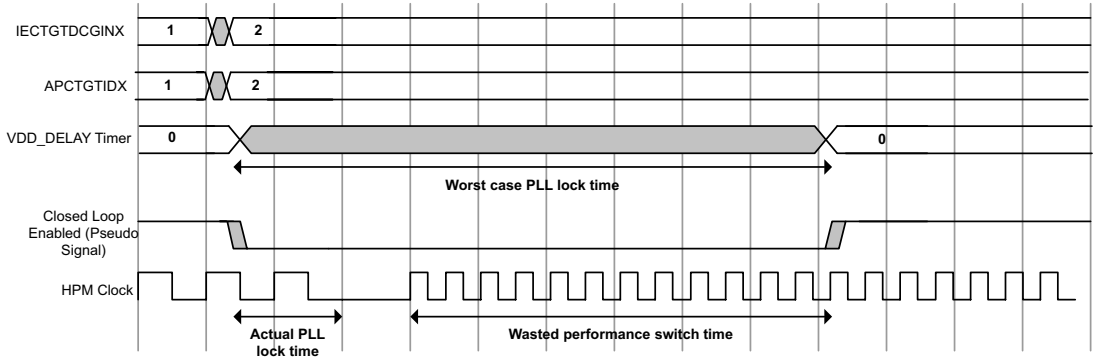

Figure 7 Standard APC1 closed-loop behavior

To solve this issue, the **Performance_Target_Index** level to the APC1 is not directly driven by the IEC. Instead, this signal is controlled by the frequency and voltage arbiter. The following sequence is used when switching to a higher performance level.

- IEM software requests a new performance level by writing to the IEC.

- The IEC updates the **IECTGTDCGIDX** and **IECTGTDVCIDX** outputs to the frequency and voltage arbiter.
- The frequency and voltage arbiter holds off the **apc_target_index** request to the APC1, and uses the PLL lock signals to accurately predict when the HPM sampling clock is ready.
- The DCG then issues the new **apc_target_index** to the APC1.
- The DCG then switches the HPM sampling clock to the new performance level.

The advantage of this system is that the timer **VDD_DELAY** in the APC1 need only be set to the maximum time taken to switch in the new HPM clock. This time is considerably smaller than the worst case PLL lock time, thus reducing any unnecessary baggage in the performance level switching. Figure 8 summarizes this operation.

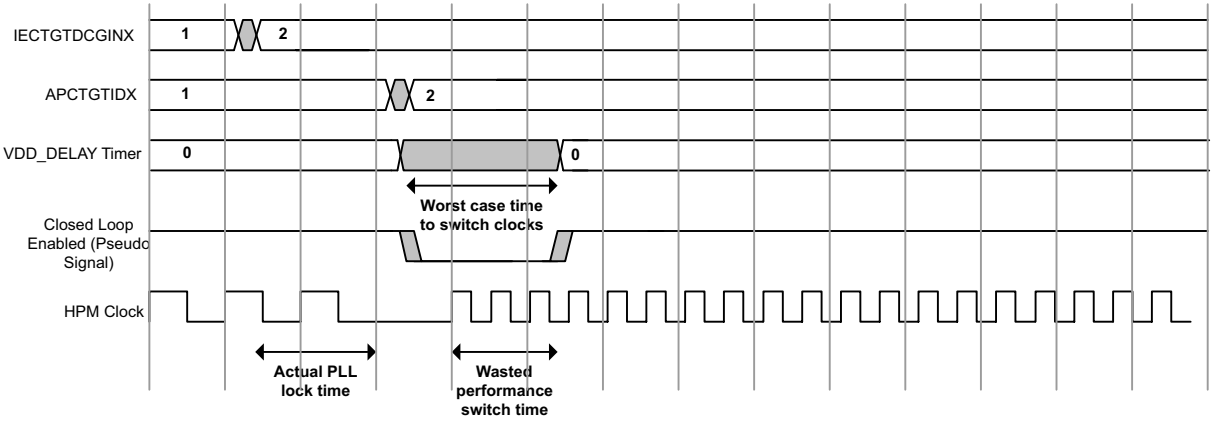

Figure 8 Improved closed-loop with target voltage under DCG control

When switching to a lower performance level, it is assumed that lower frequencies never require a higher voltage than is currently available, and that a fall in voltage to the required level while the circuit is active does not affect the circuit operation. **APC_CURRENT_INDEX** is therefore immediately updated by the APC1 to reflect **apc_target_index**. This enables the DCG to immediately switch both the system and HPM sampling clock to the lower frequency. Feedback from the HPM is then used by the APC1 to lower the voltage.

Like the open-loop mode, if the IEC issues a new performance targets before the current target has been achieved, this overrides the current target. Unlike open-loop mode which can be interrupted at any stage up until when the command to switch clocks is issued, in closed-loop mode, when **apc_target_index** has been updated to the APC1, and the new HPM clock has been selected, the system is committed to complete the

initial performance transfer. For this example 0% power-down modes are not considered. For more information on these, refer to *Dormant and Shutdown Mode* on page 24.

Performance Requirement Optimization

When operating in open-loop mode, to improve efficiency for increases in performance, the IEM control system might operate in Performance Requirement Optimization mode.

Performance Requirement Optimization can be achieved when a new performance request is issued that spans a number of the voltage index points. As each intermediate voltage point is achieved, the ARM1176 clock can be increased to the maximum voltage supported at each of these performance levels.

Figure 10 on page 18 shows an example of how the system frequency might be increased at each intermediate voltage step until the target voltage is achieved.

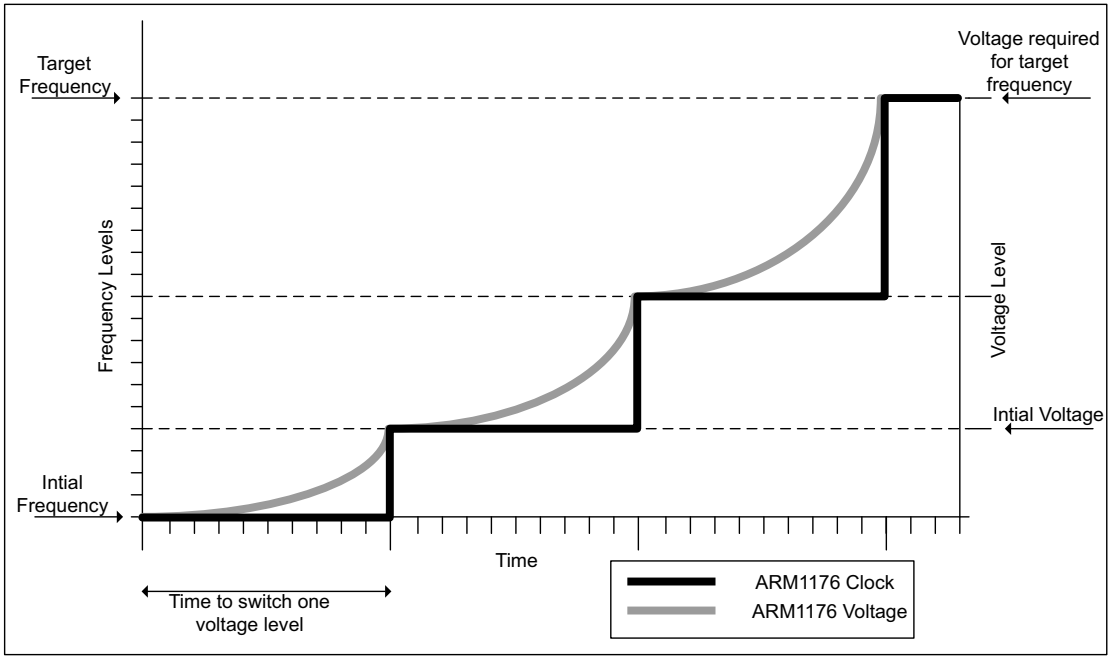

Figure 9 Performance Requirement Optimization switching

Performance Requirement Optimization requires that the DVC monitors and reports all of the intermediate voltage steps towards the target frequency. This is not the case for APC1. Instead, when the APC1 receives a new target performance level, it starts a

timer that is set to the worst case time taken to switch between the minimum and maximum operating voltage levels. When this timer has expired, the current index level is updated with the requested target level. The timer is programmed using the **VDD_DELAY** register in the APC1. Performance Requirement Optimization is therefore not implemented on the ARM1176JZF-S development chip.

2.4 IEM Slices and Enable Control

To ease implementation and to allow the ARM1176JZF-S core to operate at any frequency for IEM, all interfaces that cross a voltage domain must operate asynchronously. To facilitate this, ARM has developed an AXI IEM Asynchronous Register Slice. The AXI IEM Asynchronous Register Slice is part of the standard ARM1176JZF-S deliverables.

Although the AXI Asynchronous Register Slices operate in an extremely efficient manner, a small increase in latency might be noticed because of the synchronization logic required for each transaction between voltage domains. For this reason the AXI Asynchronous Register Slices have a synchronous mode.

Synchronous mode can only be used when the ARM1176JZF-S development chip and the rest of the SoC are operating at voltages that have been designed to allow the AXI interface on the ARM1176 development chip to operate synchronously with the on-chip AXI sub-system. This is typically when the ARM1176 development chip and SoC are operating at the maximum voltage level.

Switching between asynchronous and synchronous mode is controlled by the **SYNCMODEREQ** input on each register slices. This input must be driven to a logic 1 to request synchronous mode, and logic 0 for asynchronous mode. The AXI Asynchronous Register Slice also has an output **SYNCMODEACK** to indicate the current operating status. Logic 1 on **SYNCMODEACK** indicates that all FIFOs within the slices have been drained, and the slice is operating in synchronous mode. Logic 0 on **SYNCMODEACK** indicates that the slice operates asynchronously.

To simplify the switching between synchronous and asynchronous mode, the IEC can be programmed to automatically request synchronous mode when running at 100% performance, and asynchronous mode when running at sub-100% performance.

The IEC also provides some protection in ensuring that a sub-100% voltage and frequency are not selected until the AXI Asynchronous Register Slice has been safely switched to asynchronous mode. Likewise, when moving to 100% performance, synchronous mode is not requested until the maximum voltage and frequency has been achieved.

The IEC has a single output **IECSYNCMODEREQ** that must be connected to **SYNCMODEREQ** on each AXI Asynchronous Register Slice, and a single synchronous mode acknowledge input **IECSYNCMODEACK**. For a typical ARM1176JZF-S implementation that has multiple AXI ports and therefore multiple AXI Asynchronous Register Slices, the method of combining the **SYNCMODEACK** signals from each slice and the connection to the IEC **IECSYNCMODEACK** input must be carefully considered.

It is required that when moving to a sub-100% performance mode, the combined **SYNCMODEACK** to the IEC does not switch to logic 0 until all AXI Asynchronous Register Slice are safely operating asynchronously. If this is not the case, the IEC can request a new voltage or frequency while an AXI Asynchronous Register Slice is still operating in synchronous mode and possibly corrupt an AXI transaction.

Also when moving to 100% performance mode, the combined **SYNCMODEACK** to the IEC must not switch to logic 1 until all AXI Asynchronous Register Slices are operating in synchronous mode.

Figure 10 shows an example of how multiple SYNCMODEACK signal might be combined to create a single synchronous mode acknowledge for use by the IEC.

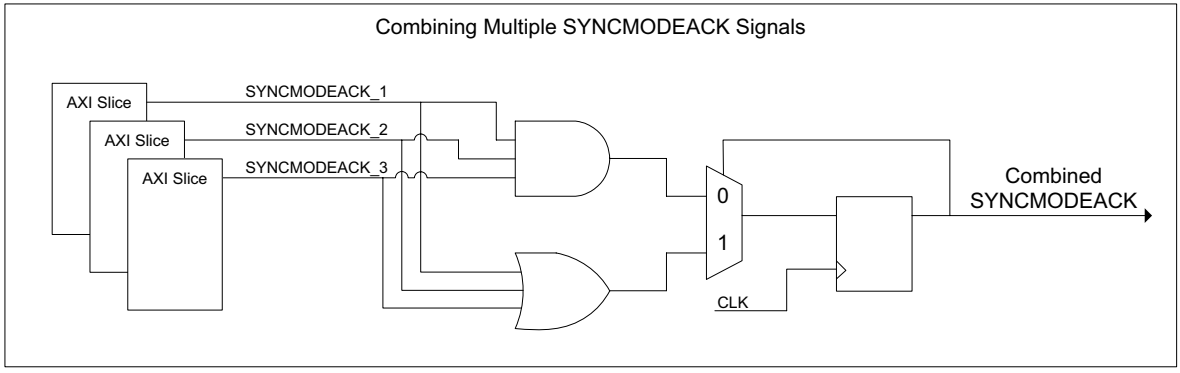

Figure 10 Combined SYNCMODEACK example

At this point, it is assumed that the reader has a clear understanding of how the clock enables are used within the ARM1176 AXI interfaces to allow for synchronous N:1 clocking of the AXI sub-system. For more information on this subject, see the ARM1176 Technical Reference Manual ref[4].

For efficient use of the AXI Asynchronous Register Slices during asynchronous mode, the clock enable **ACLKEN** must be set to logic 1. No extra complication to the system design exists if either the IEM subsystem operates in asynchronous mode, or the AXI subsystem runs at a ratio of 1:1 to the ARM1176JZF-S. In this case, the enable is always held at logic 1.

However, this is not normally the case, and in a typical system during synchronous mode, the AXI subsystem might run at a ratio of 2:1 or greater to the ARM1176JZF-S. A clock enable is therefore required to qualify transfers. When the AXI IEM Slices are in asynchronous mode the clock enable must be held HIGH. However, switching of the clock enable must be performed in a very careful manner so as not to accidentally enable invalid transfers when in synchronous mode.

Control of the enable is determined by the state of **IECSYNCMODEREQ** from the IEC, and the combined **SYNCMODEACK** signals from each IEM AXI Slice in the design. Because of the operating nature of the register slices, it is not possible to determine exactly when they enter synchronous mode following the assertion of **IECSYNCMODEREQ**, or likewise when they enter asynchronous mode following the de-assertion of **IECSYNCMODEREQ**.

Although **IECSYNCMODEACK** indicates when the transition has safely completed, internally the point at which the slices actually switch mode is indeterminate. During this undefined operating state of the slices, and to prevent any data corruption, the assumption must be made that they are running in synchronous mode. Therefore, the enable must be allowed to toggle normally to avoid data transfer corruption.

Figure 11 shows how the enable must be controlled when switching between synchronous and asynchronous modes on the AXI Asynchronous Register Slices when using N:1 clocking.

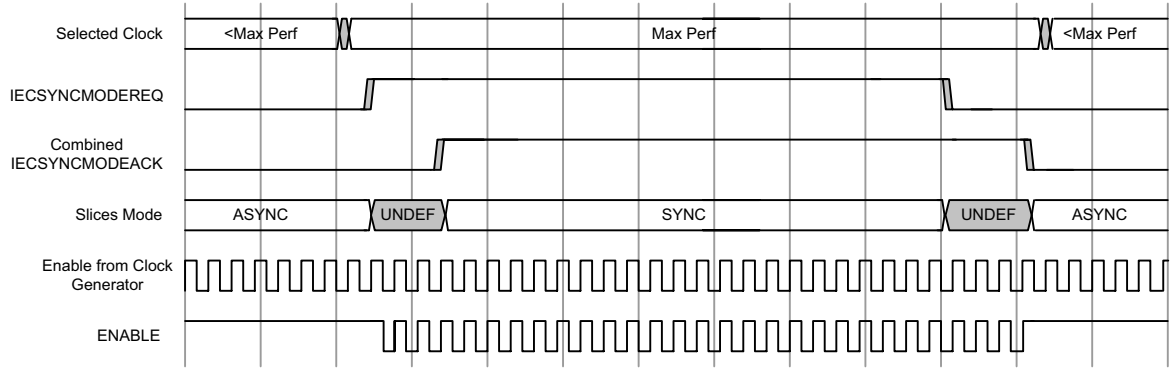

Figure 11 Clock Enable control between synchronous and asynchronous modes

Therefore the only time that the slices are in asynchronous mode, and that the enable is driven HIGH, is when both **IECSYNCMODEREQ** and **IECSYNCMODEACK** are LOW. Figure 12 shows an example of how the **ACLK** enable from a clock generator can be conditioned. This is achieved by using the **SYNCMODEREQ** signal from the IEC, and the combined **SYNCMODEACK** signal from each AXI Register Slice to safely control the **ACLK** enable.

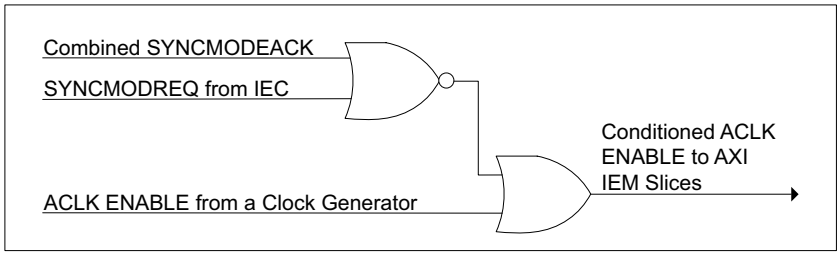

Figure 12 Example clock enable control circuitry

IEC Maximum Performance Request

For a number of system scenarios the IEC has a maximum performance request input **IECMAXPERF**. This allows the software programmed performance level to be overridden at any time and forces the IEC to request 100% performance mode.

A number of control signals can be combined to create the single maximum performance request to the IEC. For the ARM1176JZF-S development chip interrupts, the debugger and an off-chip user input can both trigger the maximum performance request.

Figure 13 on page 21 shows how special care must be taken if the sleep interrupt IECCPUSLPINT is used on the IEC. This interrupt indicates that a 0% performance request has been made to the IEC, or that the ARM1176JZF-S development chip must enter STANDBYWFI during DVS emulation mode.

It is important that this interrupt is masked to the **IECMAXPERF** input to avoid the IEC being forced to request maximum performance when 0% is required. This issue is highlighted in the IEC Errata notice document.

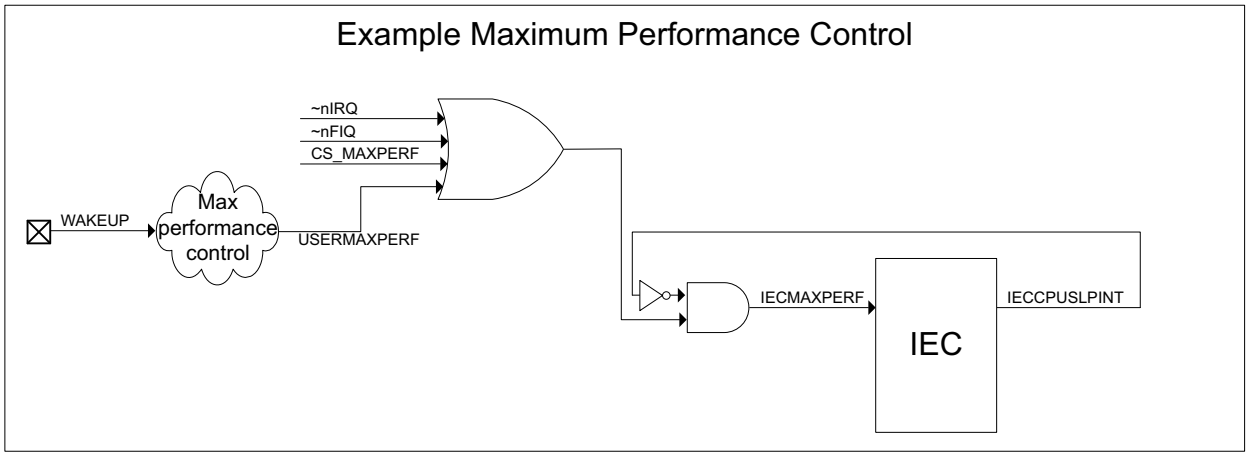

Figure 13 Example maximum performance control

Important - When **IECMAXPERF** is asserted, it does not override the programmed performance level within the IEC. After **IECMAXPERF** is deasserted, the IEM control system reverts to the previously programmed performance level.

For this reason, all maximum performance sources on the ARM1176JZF-S development chip are sticky and must be cleared by software. This enables a new performance level to be written to the IEC prior to the signal being deasserted.

Table 3 shows an example of some of the wakeup sources implemented on the ARM1176JZF-S development chip.

Table 3 IEC Maxperf / Wakeup sources

Signal	MAXPERF functionality
nIRQ	Asserted when **nIRQ** is active-LOW. Interrupt source in either GIC or TZIC must be cleared.
nFIQ	Asserted when **nFIQ** is active-LOW. Interrupt source in either GIC or TZIC must be cleared.
USERMAXPERF	Asserted by the rising edge of SoC primary input **WAKUP**. Cleared by writing zero to the System Control Register bit [4].
CS_MAXPERF	Asserted by CTI #1 **TRIGGEROUT**[1]. Enables this trigger to act as a parallel trigger to **EDBGRQ**. The debugger must perform the IEC performance write and clear this trigger.

3 IEM Power Modes

The ARM1176JZF-S development chip supports four power management modes:

- IEM Run mode
- StandbyWFI mode
- Shutdown mode
- Dormant mode

3.1 IEM Run Mode Performance Level Controller State Machine

This is the normal mode after reset, and is the only mode in which the IEM software operates. After reset, the AXI slices operate asynchronously and the IEC does not generate an **IECSYNCMODEREQ** until configured. Additionally, after reset the IEC defaults to requesting 100% performance target. In Run mode, the IEM software does not select a performance level of 0%. A performance level of 1% is the minimum demand the IEM software can set.

If the IEM software requests a new performance level before the current target has been reached, the IEC immediately updates the target performance indexes **IECTGTDCGIDX** and **IECTGTDVCIDX**. Likewise, the Run Mode Performance Level Controller State Machine must be designed to support the overriding of target performance requests.

ETM and Level 2 Cache Controller Considerations

When entering StandbyWFI, Shutdown or Dormant mode, other components closely coupled to the ARM1176JZF-S core using a common clock and shared dynamic voltage domain must be carefully considered.

The main focus is to ensure that all AXI and ETM transactions have completed before the clock is gated or power removed. This is easily done with the ETM as it automatically handshakes with the ARM1176JZF-S core using **WFIPENDING** and **nETMWFIREADY**. Therefore, the ARM1176JZF-S core holds off asserting **STANDBYWFI** until the ETM11CS has safely entered the StandbyWFI state.

However, the Level 2 cache controller does not use the same handshake procedure. This issue is solved by using a combination of a software and hardware work around.

Before issuing the CP15 WFI instruction, a cache synchronous request must be issued to the L2 cache controller. Following the cache synchronous, the **L2CCCACTIVE** and **L2CCCSYSACK** signals must be monitored until they both go LOW to ensure all AXI activity has completed. It is then safe to power down or gate the clock.

StandbyWFI Mode

StandbyWFI mode is a standard power saving feature of the ARM1176 core, allowing the clock to disable a majority of flip-flops in the core when no CPU processing is required.

In a standard ARM1176JZF-S implementation, StandbyWFI mode is entered by executing the CP15 WFI Instruction. Although this is still possible while using IEM, the IEC is unaware that the core has entered StandbyWFI and therefore, data required for the IEM software accumulated from Dynamic Performance Monitor within the IEC becomes inaccurate.

Therefore, when using IEM, StandbyWFI must be entered by first requesting 0% performance from the IEC. The IEC then generates a sleep interrupt by asserting **IECCPUSLPINT**. The service routing for this interrupt prepares the system for StandbyWFI and executes the CP15 WFI Instruction.

When entering StandbyWFI mode, no new voltage requests are made to the APC. Voltage is therefore held at the current level prior to requesting 0% performance. If closed-loop is enabled the clock to the HPM is not interrupted, and closed-loop voltage control continues to operate as normal.

Example StandbyWFI Entry and Exit Procedure

To safely enter StandbyWFI mode, the following procedure must be followed:

- Set **APC_PWRDN_EN** bit [3] to 0 in the APC_CONTROL register. This prevents the APC1 from issuing a SLEEP command on the PowerWise interface when a performance level of zero is requested. Instead, the voltage is held at the current programmed level.
- Ensure **IEC Max Perf Enable** bit [2] in the IEC DPC Control Register is set to 1 to enable the system to wake up from StandbyWFI when **IECMAXPERF** is asserted because of an interrupt or other maximum performance request.
- Write 0% performance to the IEC. This ensures that the dynamic performance monitors are frozen during StandbyWFI.
- The IEC then issues its sleep interrupt by asserting **IECCPUSLPINT**. It also requests the minimum IEM operating performance level to save energy while waiting for the ARM1176-JZF development chip to enter StandbyWFI. For example, a system with 3 performance levels of 100%, 80% and 75%, the minimum performance level of 75% is selected.
- If a Level 2 cache controller is implemented, a cache synchronous request must then be issued to the L2CC.

- Issue a CP15 WFI Instruction.
- After the ARM1176-JZF core has entered StandbyWFI mode, the **STANDBYWFI** output is asserted.
- When the IEC determines that the ARM1176-JZF core has safely entered StandbyWFI (the **STANDBYWFI** signal must be connected to the IEC input **IECCPUWFIACK**), a 0% performance request is issued.
- The **apc_target_index** is then set to 0 and the APC1 updates apc_current_index. Because the **APC_PWRDN_EN** bit [3] is set to 0 in the **APC_CONTROL** register, the voltage is not updated by the new target request.
- At this point you might choose to completely disable the clock to the ARM1176-JZF core and ETM. However, if a Level 2 cache controller is implemented, the clock must not be disabled until both **L2CCCACTIVE** and **L2CCCSYSACK** from the L2CC are held LOW indicating that all L2CC AXI transactions are complete.

The following events occur when exiting StandbyWFI mode:

- An interrupt or debug request creates an **IECMAXPERF** event.
- IEC requests 100% performance.
- Maximum voltage is returned to the system and the clock is not gated.
- The ARM1176-JZF core can now use the standard return from the StandbyWFI wake-up mechanism.
- System returns to run mode.

Important - If the maximum performance condition is removed from the IEC it reverts to the currently programmed performance level, which in this case is 0%. If this action is not required, a new performance level must be programmed.

Dormant and Shutdown Mode

During Shutdown mode, the IEM subsystem of the ARM1176-JZF development chip is completely powered-down. This includes the ARM1176-JZF core, the L2CC, and the ETM11. During this mode it must be ensured that outputs from the IEM subsystem are clamped at safe levels to prevent any unwanted behavior towards the rest of the SoC.

The ARM1176-JZF development chip also supports Dormant mode. During Dormant mode, the state of the Level 1 cache and TCM RAMs within the ARM1176-JZF core are retained. To save the maximum amount of energy, the RAMs are also held at a very low retention voltage. This voltage only requires to be sufficient to maintain state within the RAM bits cells, and is probably too low for the RAM to operate correctly.

During Dormant mode on the ARM1176-JZF development chip, the status of the Level 2 cache RAMs within the L2CC are lost. It is therefore imperative that a Level 2 cache synchronous write-back is performed before entering Dormant or Shutdown mode.

The LP5550 has four operating states, STARTUP, ACTIVE, SLEEP and SHUTDOWN. ACTIVE state is used for normal IEM mode and scales both the core and RAM voltage regulator in step. During Dormant mode the LP5550 is switched to SLEEP state by the APC1 issuing a sleep command request on the PWI. During this state the core voltage (VFB) is reduced to zero and the RAM voltage (VO3) to the programmed retention level. The LP5550 can then return from SLEEP by issuing a wakeup command request on the PWI.

However for shutdown mode the APC1 still requests that the LP5550 enters SLEEP state. Therefore, the RAM voltage (VO3) is only reduced to the programmed retention level. To truly benefit from the energy savings of Shutdown mode when using the LP5550, a mechanism must be implemented to power gate the RAM voltage (VO3) supply.

Depending on the system implementation, it might be possible to use the SHUTDOWN state of the regulator for Shutdown mode. However, SHUTDOWN state completely powers down all of the LP5550 supplies. Therefore, both the fixed voltage (VO1) and the I/O power supply (VO2) are powered down along with the core voltage (VFB) and RAM voltage (VO3).

In addition, the APC1 cannot return the LP5550 to ACTIVE state from SHUTDOWN by issuing a wake-up command on the PWI. Return from shutdown can only be achieved by toggling the **ENABLE** or **RESETN** inputs.

Example Shutdown and Dormant mode Entry and Exit Procedure

During the Shutdown or Dormant mode, all state within the ARM1176-JZF core is lost. Therefore, only a reset can initiate a return from one of these power-down modes. It is the task of additional software to report the system status prior to reset. This might be a normal power-on reset, software reset, shutdown or Dormant mode reset.

In the ARM1176-JZF development chip, this is achieved using a customized system controller. Before entering shutdown or dormant mode, the **SHUTDOWNNXT** or **DORMANTNXT** flags are set respectively in the System Controller Control Register. Software might then read these flags when returning from a power-down mode. These

flags also control the **SHUTDOWN** and **DORMANT** primary outputs on the ARM1176-JZF development chip. The relevant output is set after the SoC has safely entered one of the power-down modes. These outputs are used for additional off-chip power gating if necessary.

The following sequence is used to enter Shutdown or Dormant mode:

- **SHUTDOWNNXT** or **DORMANTNXT** flag is set in the ARM1176-JZF development chip System Controller Control Register.
- System software (not IEM driver) requests 0% performance level by a write to the IEC.
- The IEC asserts **IECCPUSLPINT** interrupt.
- On receiving the IEC **IECCPUSLPINT** interrupt, the ARM1176-JZF core disables all interrupts, flushes write buffers, and allows DMA channels to drain. The ARM1176-JZF core also performs a handshake with **ETM11CS** to ensure all trace buffers have been drained.
- **APC_PWRDN_EN** and **ACP_VDD_UD** bits are set to 1 in the APC_CONTROL Register. It does not matter whether the APC1 is in open- or closed-loop when Shutdown or Dormant mode is entered or exited.
- Ensure bit [2] of IEC Max Perf Enable in the IEC DPC Control Register is set to 1. This allows the system to wake up from Shutdown or Dormant mode by the assertion of **IECMAXPERF** input on the IEC.
- **IECCPUSLPINT** interrupt is cleared.
- The CP15 Wait For Interrupt (WFI) instruction is issued.
- The ARM1176-JZF core then enters StandbyWFI and asserts **STANDBYWFI**.
- On determining that the ARM1176-JZF core has safely entered StandbyWFI, the IEC sets **IECTGTDCGIDX** and **IECTGTDVCIDX** to an index of 0. At this point depending on your system, you might or might not want these 0% performance requests to instantly be forwarded to both the DCG and APC1.
- If you have a Level 2 cache controller you must wait until both **L2CCCACTIVE** and **L2CCCSYSACK** from the L2CC are held LOW, indicating that all L2CC AXI transactions are complete.
- When all activity has ceased, the 0% performance request from the IEC is then forwarded to the DCG and APC1 using the frequency and voltage arbiter.
- The DCG asserts a reset on the ARM1176-JZF core and other IEM subsystem components that might include the ETM11CS and L2CC.

- On seeing a target performance request of 0%, the APC1 asserts the **apc_clamp_req** signal, and if in closed-loop mode, the loop filter is disabled.
- On seeing the **apc_clamp_req**, the I/O clamps must be enabled around the IEM subsystem, and L1 cache RAMs if entering dormant mode.
- After the IEM subsystem is safely clamped, **apc_clamp_ack** must be asserted to the APC1.
- The APC1 issues the sleep command to the regulators using the PowerWise Interface and IEM subsystem voltage is driven to zero.
- The rest of the SoC continues to the powered and clocked as normal.

Returning from Shutdown or Dormant Mode

To exit Shutdown mode, a maximum performance condition must occur.

- The maximum performance condition causes **IECMAXPERF** to the IEC to be asserted. The IEC then requests maximum voltage and frequency to the frequency and voltage arbiter.
- On seeing **IECTGTDCGIDX** and **IECTGTDVCIDX** set to maximum, the frequency and voltage arbiter sets the **apc_target_index** to the maximum level on the APC1.
- The APC1 issues the wakeup command to the regulators using the PowerWise Interface.
- APC1 then polls the **COREVDDOK** status of the regulator using the PowerWise Interface after the internal wakeup timer has timed-out.
- When APC1 confirms that the **COREVDDOK** status is OK, the **apc_current_index** is updated to maximum performance.
- The APC1 deasserts **apc_clamp_req**. I/O clamps around the IEM subsystem must now be disabled.
- When the full voltage has been returned, the frequency and voltage arbiter requests the maximum performance clock from the DCG. If the APC1 is in closed-loop mode, the HPM clock must also be started.
- To support TrustZone the memory map must be restored to the power-on reset settings. See *TrustZone* on page 29.
- The DCG then releases reset to the IEM Subsystem.

- The system boot code must read the power-down status flags to determine the cause of the last reset.
- Important - If the maximum performance condition is removed from the IEC it reverts to the currently programmed performance level, in this case 0%. If this action is not required, a new performance level must be programmed. It is suggested that following a power-down, the system boot code must always program the IEC with maximum performance.
- Clear the maximum performance condition.
- System is restored to pre-power down state.

4 TrustZone

The ARM1176JZF-S processor, and associated peripherals and fabric, implement TrustZone technology. This requires some consideration from the system design perspective.

Whenever the core exits a reset state it boots in secure state, and must be provided with a suitable secure boot image - typically residing in SoC ROM. Therefore, whether this is a cold power on, or exit from Shutdown or Dormant mode, secure ROM must be remapped into the CPU instruction and data port memory maps at the reset location. Clearly this is simply an extension of any existing memory remap logic, but forms part of the requirement of the controlling logic for assertion and release of signal clamps, resets, clocks and power supplies.

Figure 14 shows the logic used for controlling remap in the ARM1176JZF-S-S development chip. The important thing to note is that remap is replicated for the core and for the other system masters that rely on any remap. Once the core has booted and remap has been cleared, the system controller must reinstate remap for secure ROM for the core,

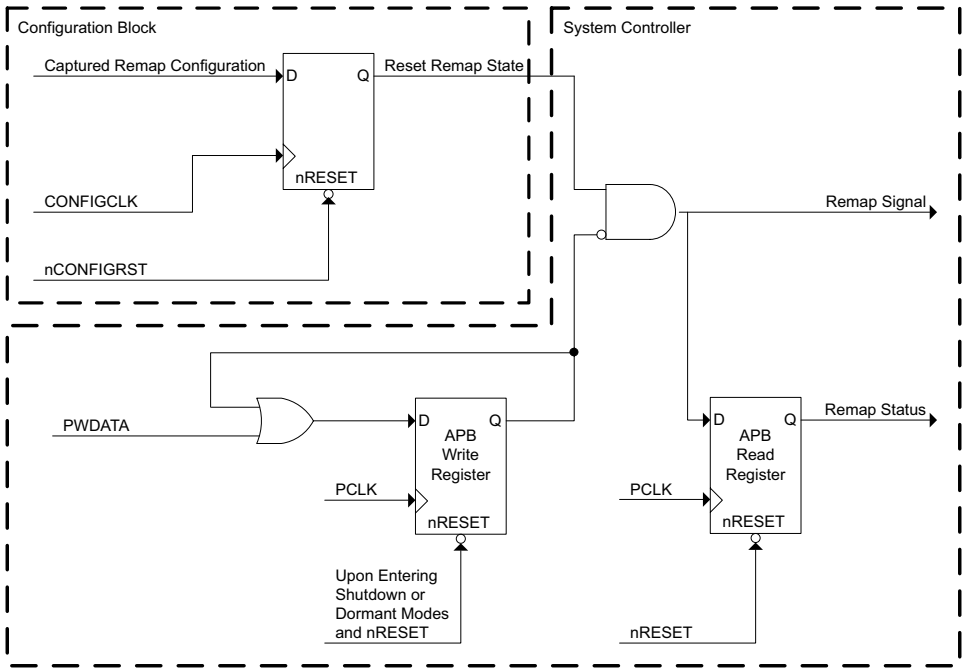

Figure 14 Remap logic

5 References

This document refers to, or should be read in conjunction with, the following documents.

Table 4

Ref	Author(s)	Title
1	ARM	Intelligent Energy Controller (IEC) Technical Reference Manual
2	National Semiconductor	Advanced Power Controller (ACP1) Technical Reference Manual
3	National Semiconductor	LP5550LQ - PWI and PowerWiseTM Technology Compliant Energy Management Unit
4	ARM	AMR1176JZF-S Technical Reference Manual