A 90-nm Variable Frequency Clock System for a Power-Managed Itanium Architecture Processor

Tim Fischer, Jayen Desai, Bruce Doyle, *Member, IEEE*, Samuel Naffziger, *Member, IEEE*, and Ben Patella, *Member, IEEE*

***Abstract*—An Itanium Architecture microprocessor in 90-nm CMOS with 1.7B transistors implements a dynamically-variable-frequency clock system. Variable frequency clocks support a power management scheme which maximizes processor performance within a configured power envelope. Core supply voltage and clock frequency are modulated dynamically in order to remain within the power envelope. The Foxton controller and dynamically-variable clock system reside on die while the variable voltage regulator and power measurement resistors reside off chip. In addition, high-bandwidth frequency adjustment allows the clock period to adapt during on-die supply transients, allowing higher frequency processor operation during transients than possible with a single-frequency clock system.**

***Index Terms*—Adaptive frequency control, clocking, clocks, delay-locked loop (DLL), microprocessor, phase-locked loop (PLL), variable frequency, voltage-frequency converter, voltage-locked loop.**

I. Introduction

THE Montecito CPU is an Itanium Architecture processor containing two cores using Foxton technology on a 1.7 billion transistor die. The cores are 6-issue two-way multithreaded and the caches in each core consist of: L1 16 K instruction and 16 K data; L2 1 MB instruction and 256 K data; unified L3 12 MB [1]. Foxton technology provides power management capability by maximizing performance within a configured power envelope. Core supply voltage and clock frequency are modulated dynamically in order to remain within the power envelope. The Foxton controller and dynamically variable clock system reside on die while the variable voltage regulator and power measurement resistors reside off-chip [2]–[4].

II. Montecito Clock System Overview

The clock system uses the input system bus clock to generate a range of fixed and variable frequency clocks. As shown in Fig. 1, the clock system consists of a single PLL which generates a multiple $6 \leq \mathrm{M} \leq 20$ of the system clock frequency. This PLL output clock, running at a frequency called Fmax, is distributed to 14 digital frequency dividers (DFDs) for division to the proper zone frequency [3]–[5].

Montecito has a total of 14 local clock zones: six variable and eight fixed frequency. The clock zones consist of: two cores each with three DFDs; one 1-GHz DFD for Foxton technology control; one DFD for each of six front side bus (FSB) stripes; one DFD for bus logic. The bus logic and FSB stripe clock zones are collectively known as the I/O clock domain and are described in more detail below.

In the core, each DFD output clock is distributed to 6–10 second level clock buffers (SLCBs) shown in Fig. 2 via level 1 (L1) clock routes, with a total of 26 SLCBs spread across three DFD zones. The SLCBs produce the final level 2 clock route (L2) which is driven to 7536 clock Vernier devices (CVDs) per core for local delay fine-tuning via scan. Gaters provide a final gain stage, power-saving enables, and pulse shaping for low latching overhead and skew compensation through transparency [6].

The I/O clock domain encompasses the six bus stripes embedded in the cache as well as the MPE protocol block in the center of the chip. A single low-skew, fixed-frequency clock domain, called the *MIB clock*, covers these seven regions. The MIB clock is an integer multiple, $6 \leq \mathrm{M} \leq 20$, of the system clock, whose frequency can range from 200 MHz to 333 MHz. The MIB clock domain uses the same circuit elements as the core clock trees. Fig. 3 shows a block diagram of the MIB clock tree. The MIB clock domain also generates and distributes (in a matched way) a 90 degree clock, called Clock90, which is unique to the six bus stripes. In normal chip operation, this clock runs at the MIB clock frequency but its phase lags by 90 degrees. Its purpose is explained in the I/O clocking section, Section IV.

III. Level 2 Distribution and Regional Active Deskew

Often, a grid is used to distribute the final level of the clock tree in order to minimize skew and simplify routing. However, a grid system is highly capacitive, costs a large amount of power, and consumes a large quantity of the coveted high level, low resistance metal resources. To minimize L2 clock route power, a simple low capacitance H-tree route was employed. H-tree systems, while being low capacitance, suffer from imbalances between branches due to design mismatch, temperature, voltage, and random and systematic process variation. To null this skew, a regional active deskew (or RAD) system was employed, as shown in Fig. 2. SLCBs are fed by the L1 routes and they produce output clocks SLCBO[1:0]. SLCBs are not located next to each other and therefore there are device, voltage supply, and temperature variations from one buffer to another. These variations, along with any skew differential in the L1 routes, manifest as skew between the SLCBO outputs.

In order to null the skew a 128 bit delay line, which has delay steps of approximately 1 ps, is incorporated into the SLCB. Two SLCBO clocks are compared by the phase comparator shown in Fig. 2(b). The comparator's output is dictated by the phasing

Manuscript received May 15, 2005; revised August 16, 2005.

The authors are with Intel Corporation, Fort Collins, CO 80528 USA (e-mail: timothy.fischer@intel.com).

Digital Object Identifier 10.1109/JSSC.2005.859879

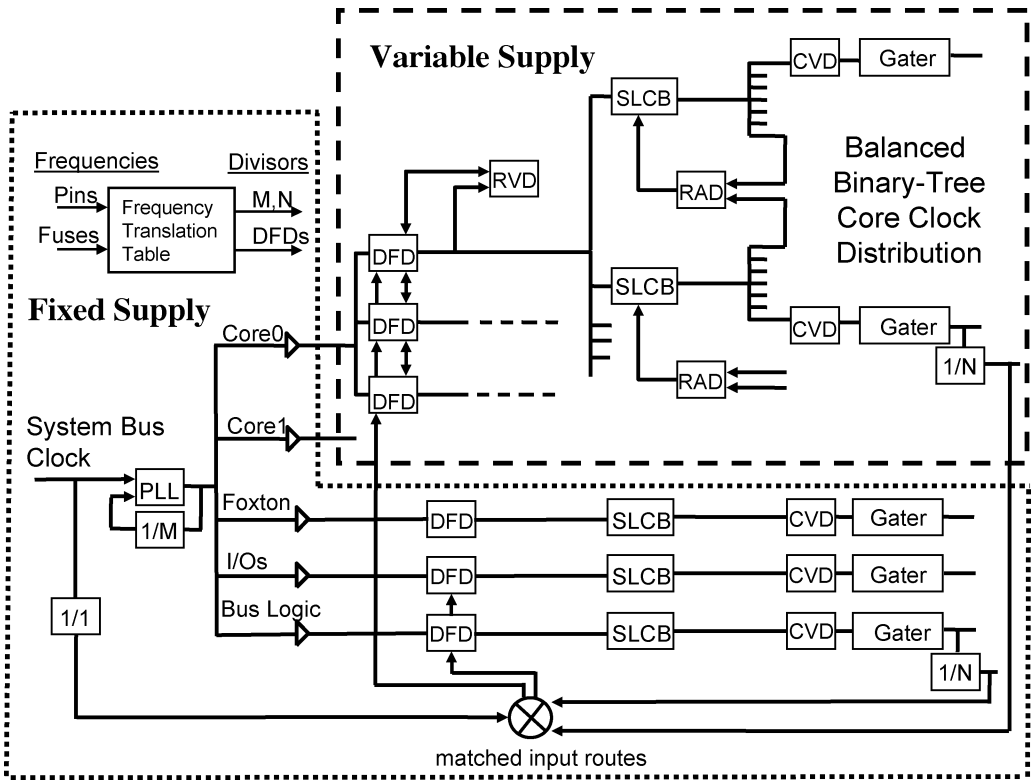


Fig. 1. Montecito clock system architecture.

of the two input SLCBOs. The comparator catches the first arriving falling edge to determine the phase relationship of the two clocks. Detection is achieved using a dynamic Set-Reset flip-flop. If SLCBO[1] lags SLCBO[0], the phase comparator outputs a high level, incrementing the delay line in the buffer that produces SLCBO[1], hence pushing SLCB0[1] closer to SLCBO[0]. Decrements are indicated when SLCBO[1] leads SLCBO[0] at the phase comparator. This continues until the two clocks are lined up. The complete RAD system is comprised of 30 phase comparators and 26 second level clock buffers in each core connected in a tree-like structure. In this structure all of the clock zones in the core are deskewed to a central anchor zone. The hierarchical nature of the system introduces a finite amount of skew but the RAD system manages this skew to approximately 10 ps across each core.

A more detailed view of the SLCB is shown in Fig. 2(a). The two half rate clocks signals enter the SLCB and are XNORed together to produce a full rate clock. The full rate clock is then driven into the 128-bit variable delay line. The delay line is comprised of a series of switchable capacitors that are driven by a current starved inverter. There are a total of eight inverters within the delay element each of which drives 16 switchable capacitors. Absolute matching of the capacitors and drive current is not critical since the system will produce aligned clocks regardless of the variability in the delay element step size. The number of capacitors switched on in the variable delay line is controlled by the SLCB state machine.

The state machine consists of an input zone summer, a digital filter, control logic, and a 128 bit barrel shifter. As shown in Fig. 2(a), the phase comparator output is received by the zone summer. The summer has two inputs, allowing for the deskewing of three zones between which data transfers are critical. The output of the summer indicates to the state machine that it need to increment or decrement the barrel shifter. The output has a scan-configurable digital filter that requires multiple increment or decrement instructions to be detected before either operation is executed. This dampens RAD response and ensures that, once the system is locked, single noise-related events do not cause the deskew system to dither unnecessarily. Once the filter allows the instruction to pass, the control block causes at barrel shift to shift up or down as required. The delay element drives a scan-controlled duty cycle adjust circuit. The duty cycle of the output clock can be controlled in a range of 40/60% to 60/40%. The desired duty cycle can be loaded uniquely into each SLCB during system boot. This feature is used for fixing phase-clocked timing paths between clock zones. Note that only one edge of the clock is manipulated to ensure the duty cycle offset does not upset clock system deskewing.

Details of the current-starved delay inverter structure, used to construct the variable delay line, are shown in Fig. 2(c). The clocks are transmitted through the delay element differentially. The internal delay inverters are connected to the rails through high impedance current mirror devices which are used to dictate the current following out of the structure into the switchable capacitors and to increase the power supply rejection ratio (PSRR). The PSRR is further enhanced by bypassing the intermediate nodes of the structure with the device pcap and ncap. These devices act to provide a momentary power source in the event of a high-frequency power supply glitch. By supporting the internal nodes of the structure the insertion delay of the SLCB, which has be set to its deskew value is not affected and hence systems transients are minimized.

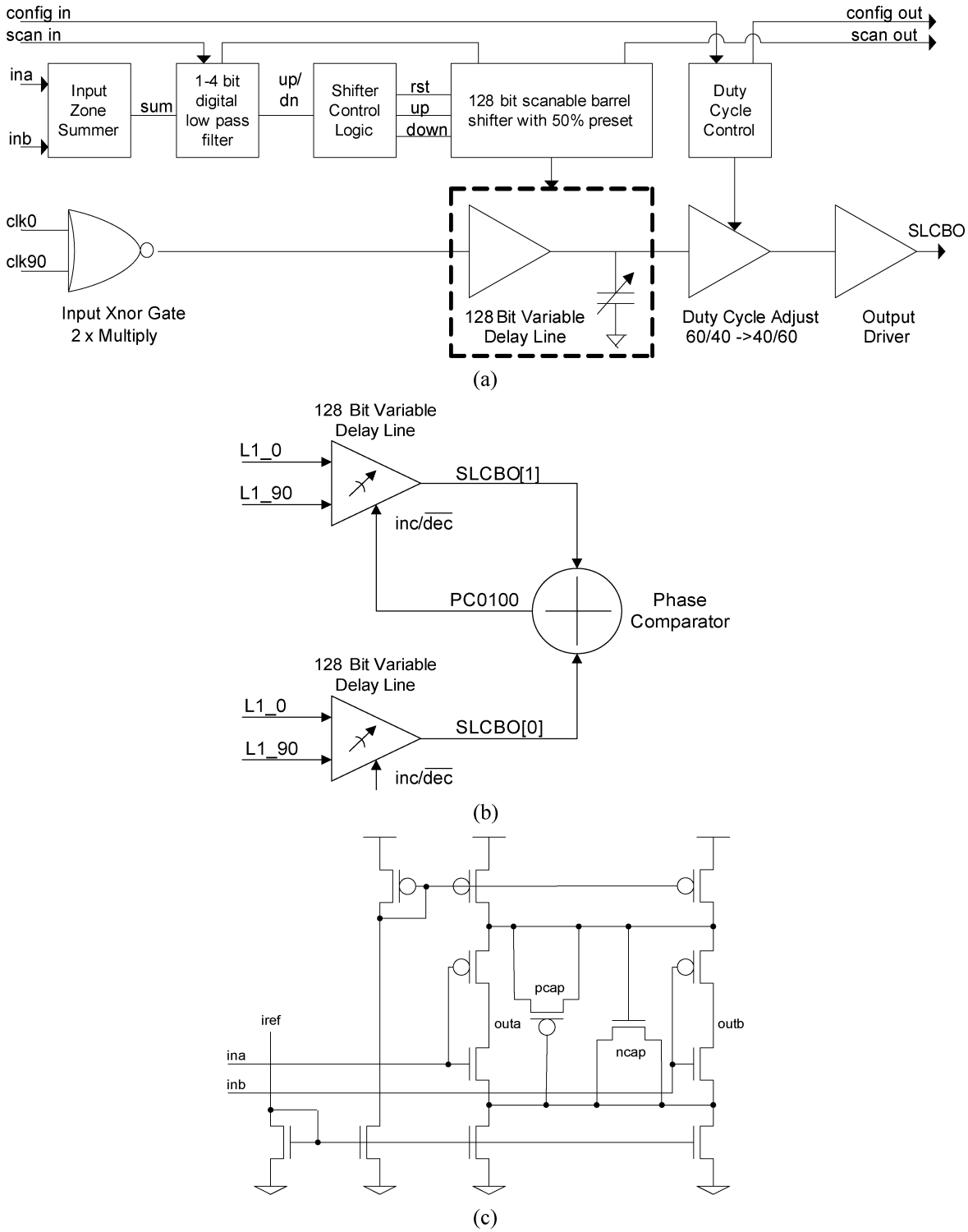


Fig. 2. Second level clock system buffer (SLCB). (a) SLCB functional diagram. (b) RAD SLCB/comparator connections. (c) Variable delay line current-starved inverter.

IV. I/O Clocking

The I/O clock domain has some special requirements. First, the MIB clock must generate the required drive, receive, data, and strobe clocks for the I/O timing protocol. This means that we must reconstruct, from the MIB clock, clock edges which occur at the 0%, 25%, 50%, and 75% points of the system clock cycle. Second, the I/O timings require that the reconstructed 0% clocks are aligned to the rising edge of the system clock at the chip inputs. Third, the I/O requires special clock test features to margin and test the I/O buffer timings.

The I/O timing protocol requires that Montecito drives and samples its common clock I/Os on the system clock rising edge. Furthermore, the source synchronous data is driven at the 0% and 50% points of the system clock period; strobes are driven at the 25% and 75% points of the system clock period. These

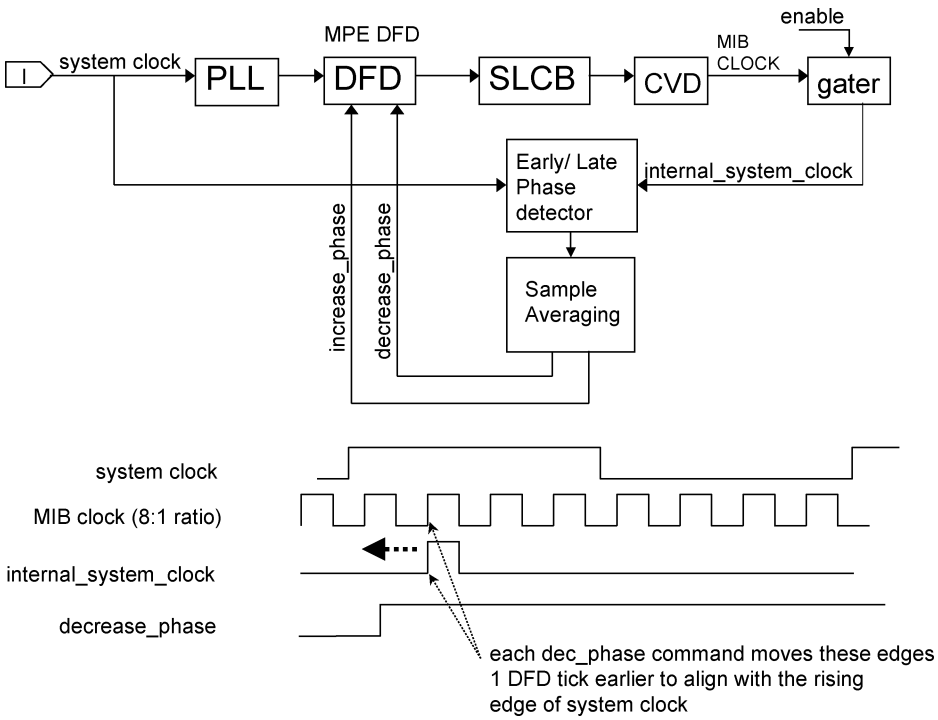


Fig. 3. System/MIB clock phase aligner block diagram and timing.

clocks are generated by selectively enabling one of the higher frequency MIB clock edges. For odd ratios of MIB to system clock, this requires the use of the clock90 path to get correct strobe timing place. A counter is instantiated in each of the six I/O stripes to select a specific MIB clock or MIB clock90 cycle for an I/O drive, receive, or strobe clock. The counters values are compared against target values for each of the clock types to decide when the reconstructed system clocks should occur; in addition, the clock gaters have the ability to be positive or negative polarity, and they can multiplex between the MIB clock or MIB clock90 as required. Full JTAG overrides are available for the counter values, clock90 selection, and clock polarity selection. This feature allows one to manipulate the reconstructed system clock edges in a characterization environment in order to stress I/O timing paths without changing the system frequency.

In order to meet the second I/O clock requirement from above, the need to align I/O clocks to the system clock inputs at the bumps, we use the ability of the DFD's state machine to achieve infinite phase rotation by selecting progressively later (or earlier) phases from cycle to cycle. A block diagram of the system clock alignment functionality is shown in Fig. 3. This aligner topology is similar to the dual delay-locked loop (DLL) system described in [7]. The sample averaging reduces the susceptibility to high frequency system clock jitter. Stability of the loop is ensured by the averaging filter depth and because the feedback loop allows one system clock cycle for newly selected DFD phases to quiesce before collecting new samples from the early/late phase detector. The same increase phase and decrease phase commands which are sent to the MPE block's DFD are also sent to the six other DFDs in the FSB I/O stripes; hence, all of the MIB clock nodes will move in concert to align to the system clock within the tolerance of open loop route and circuit matching.

V. Clock System Startup

Because the clock system supports a range of system bus frequencies from 200 to 400 MHz as input and provides a wide range of fixed and variable clocks to the CPU, it must be configured at startup to set PLL and DFD divisors and operating ranges. To accomplish this, a "translation table" determines PLL, DFD, and aligner divisors from pin-selected system clock frequencies (200, 266, 333, and 400 MHz) and fuse-selected I/O and core clock frequencies. Fuses determine the core startup ("safe") and limit frequencies. The clock system has two frequency modes: *fixed* and *variable* (FFM and VFM, respectively). The clock system always starts in FFM and is then placed into VFM by firmware.

In FFM, 13 of the 14 DFDs (core and I/O) are frequency and phase aligned; the 14th is always a fixed 1 GHz for Foxton controller power management algorithms. The 13 aligned DFDs have identical fixed divisors: 0 for maximum FFM frequency, and >0 to achieve a "safe" startup frequency before entering VFM. On powerup or reset, DFD DLLs start and lock on the PLL clock autonomously. Once all 13 DLLs lock, DFD dividers start synchronously and remain phase/frequency locked to the PLL clock.

In FFM, the core and I/O clocks align to the external system clock by a phase aligner system. This aligner adjusts DFD phase selection using up/down controls, sliding the phase around without changing frequency. At startup the aligner eliminates built-in core/FSB route mismatch and aligns both to the system clock to within 20 ps across PVT. DFD clock synthesis allows phase adjustment in uniform 1/64 cycle steps with virtually infinite range.

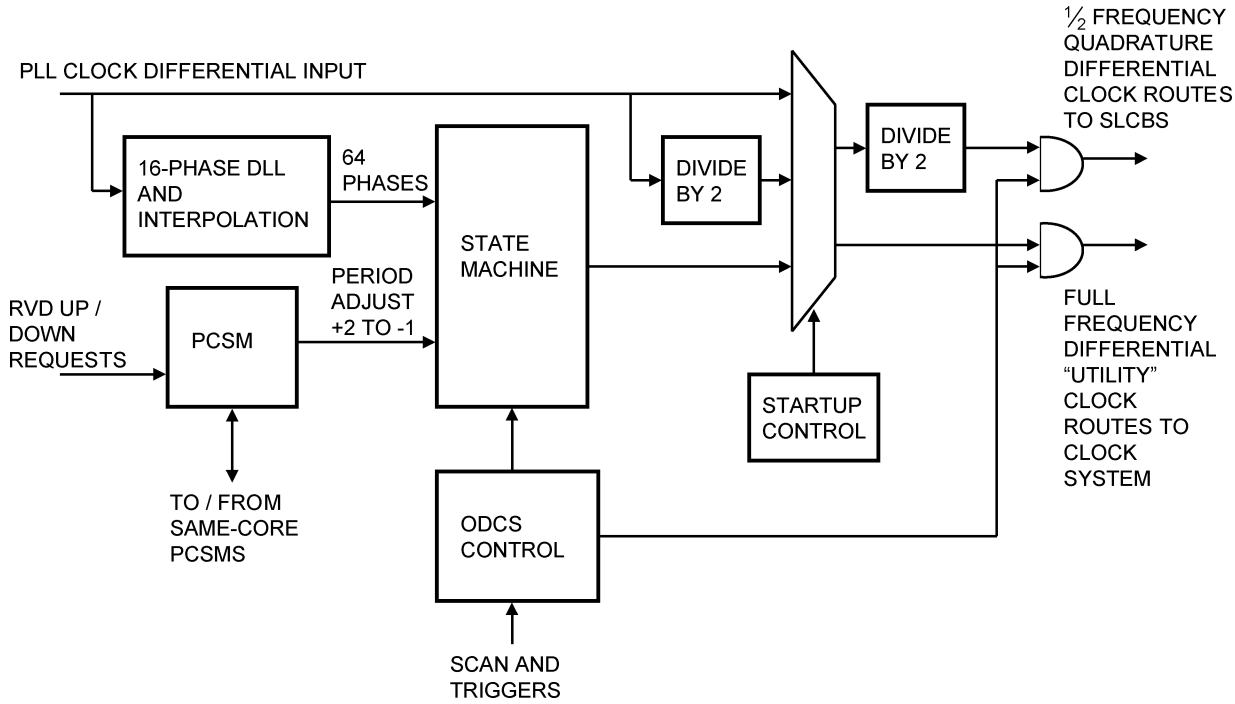


Fig. 4. DFD/PCSM block diagram.

VI. Digital Frequency Divider

The DFD is shown in Fig. 4. The DFD produces an output clock frequency based on the incoming PLL clock frequency and the current divisor value held in the phase control state machine. The output frequency range is bound by

$$\mathrm{Fdfd_{max}} = \mathrm{Fpll}$$
$$\mathrm{Fdfd_{min}} = \mathrm{Fpll} * \left(\frac{64}{(64+\mathrm{D})}\right).$$

The self-biased DLL receives differential PLL clock inputs and generates 16 evenly-spaced phases over the PLL cycle. The 16 phases are interpolated in two stages to 64 phases and the matched and balanced phase clocks are routed to a final phase select multiplexer which dynamically selects among the 64 DLL phases. This allows the DFD output frequency to vary according to $F_{\mathrm{DFD}} = F_{\mathrm{PLL}}/(1+\mathrm{D}/64)$ for divisors $0 \leq \mathrm{D} \leq 63$, yielding a range of 1.0 to $0.504 * F_{\mathrm{PLL}}$ in 1/64th increments (known as *ticks*). For a divisor of 0 the DFD clock runs at frequency Fpll. For divisors other than 0 the phase mux synthesizes a clock frequency by selecting the appropriate edges from the 64 phases, according to the equation

$$\text{next phase} = \text{modulo} - 64(\text{current phase} + \text{divisor} + \text{offsets}).$$

For example with a divisor of 1 the first rising edge of the clock would be phase N. The accompanying falling edge would be $\mathrm{N} + 33$. The second rising edge would be the modulo-64 value of $\mathrm{N} + 65$, or $\mathrm{N} + 2$. This yield a clock whose period is $\mathrm{Tmax} + \mathrm{Tmax}/64$.

Phase offsets are controlled by one or more of four phase offset sources: 1) voltage-frequency converter; 2) on-die clock shrink (ODCS); 3) phase aligner; 4) AC I/O loopback offset. In VFM, core phase alignment is shut off. Also in VFM, since ODCS adjustment is nondeterministic and difficult to measure, it is typically not used. The AC I/O loopback offset source is driven from dedicated I/O test logic and us used to step through 0/90 degree phase offsets during manufacturing test to determine I/O setup/hold margins.

A bypass mux in implemented within the DFD to allow the PLL clock to bypass the DLL during boot while the DLL is attempting to lock. In this mode, the bypassed PLL clock is divided by two to ensure that at all PLL frequencies the clock sent to the core is within the frequency operating range of the logic.

The DFD output clock are differential to allow for high-speed low noise level 1 routes. In the core the routes were of a length such that clocks could not be driven at the maximum required core frequency. In order to hold the number of DFDs in the core to three, half-frequency 0 and 90 degree clocks are transmitted to the SLCB where they are multiplied together to produce a full rate clock. A 90 degree clock was easily attained with the DFD design by transmitting clocks constructed of phases that are separated by 16. Since it has access to all 64 phases of the input clock, the DFD state machine (for those DFDs inside the I/O stripes) generates another clock called MIB clock90, which is normally generated so that it leads MIB clock by 90 degrees of the MIB clock period. Use of this clock is described in the I/O clocking section of this paper.

VII. Variable Frequency Mode and Voltage-to-Frequency Converter Design Tradeoffs

Variable frequency mode (VFM) is used to allow core clock frequency to track supply voltage changes during Foxton power modulation. Several design options were considered for the frequency control mechanism, including the use of: 1) a voltage versus frequency lookup table; 2) a core voltage based VCO;

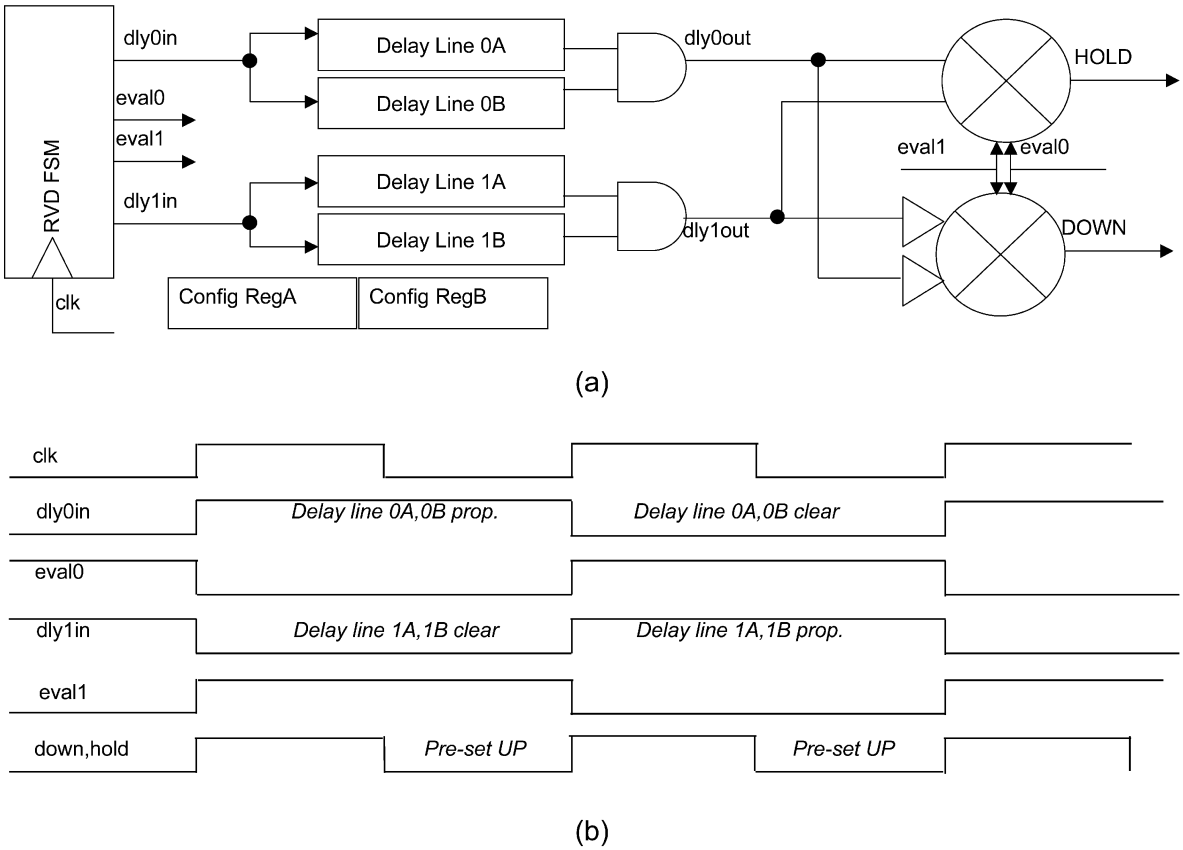


Fig. 5. Regional voltage detector details. (a) RVD architecture block diagram. (b) RVD internal timing.

3) discrete coarse-grained steps. These were all rejected in favor of our existing approach for several reasons.

A voltage versus frequency table must be calibrated at manufacturing test to compensate for the wide range of critical path V–F curve (shmoo) shapes across parts, since a table of static, design-time constants introduces too much inefficiency. Second, sensor-based table lookup requires a central voltage comparator and frequency controller which slows response to voltage changes, leaving the design exposed to voltage transients.

Use of a voltage-controlled oscillator (VCO) running off the filtered core supply has the attraction of simplicity of operation, but the need to support synchronous core clocking requires a single clock source, and thus, a single VCO. Since the core spans over 12 mm on a side, the time to detect a voltage transient and adjust the local clock frequency requires 1.5 transits plus compute time, too late to handle first droop events.

Finally, there is the "gear ratio" concept employed in PowerPC 970+ where 2–3 frequencies can be selected via a multiplexer [12]. This has the benefits of simplicity and determinism, but suffers from the same slow first droop response as table and central VCO techniques. Also, the coarse grained steps force a waiting period which wastes V^2 power and effectively limits the frequency for V/F adjustments. Step coarseness induces an average 1/2-step error which introduces further inefficiency. For example, with 100 mV/200 MHz steps, the average error would be 50 mV resulting in 10% wasted power.

To overcome these response and efficiency issues, Montecito implements a distributed delay line voltage-to-frequency converter (VFC). It allows single-cycle response to local voltage transients to reduce first droop magnitude and expensive voltage guardband. It uses digital controls for frequency adjusts, range limits, and sophisticated, repeatable clock manipulation. Finally, it provides frequency steps of 1.5%—the same order of granularity as the 12.5 mV resolution of the Foxton-regulated power supply.

Next, voltage sensors used in the Montecito VFC are described and the Montecito VFC is described in more detail.

VIII. Regional Voltage Detector (RVD) Implementation

The RVD design meets the following specifications: 1) delay matching of FET and RC-dominated paths (with up to 30% RC content) with no more than 2% error over the 0.8 to 1.2 V core supply range; 2) scan-configurable delay lines providing a delay range of 350 ps to 1 ns at typical process, 1.1 V, 90 °C; 3) one evaluation per clock cycle bandwidth; 4) deadzone with 0–50 ps configurability.

The RVD architecture, shown in Fig. 5, consists of four delay lines, a small finite state machine, two banks of scan-programmable configuration registers, and two multiplexing phase comparison latches.

The four delay lines in Fig. 5(a) are configured on a per-part basis to match Montecito critical paths in delay and voltage sensitivity. Delay lines 0A and 1A are configured by configuration

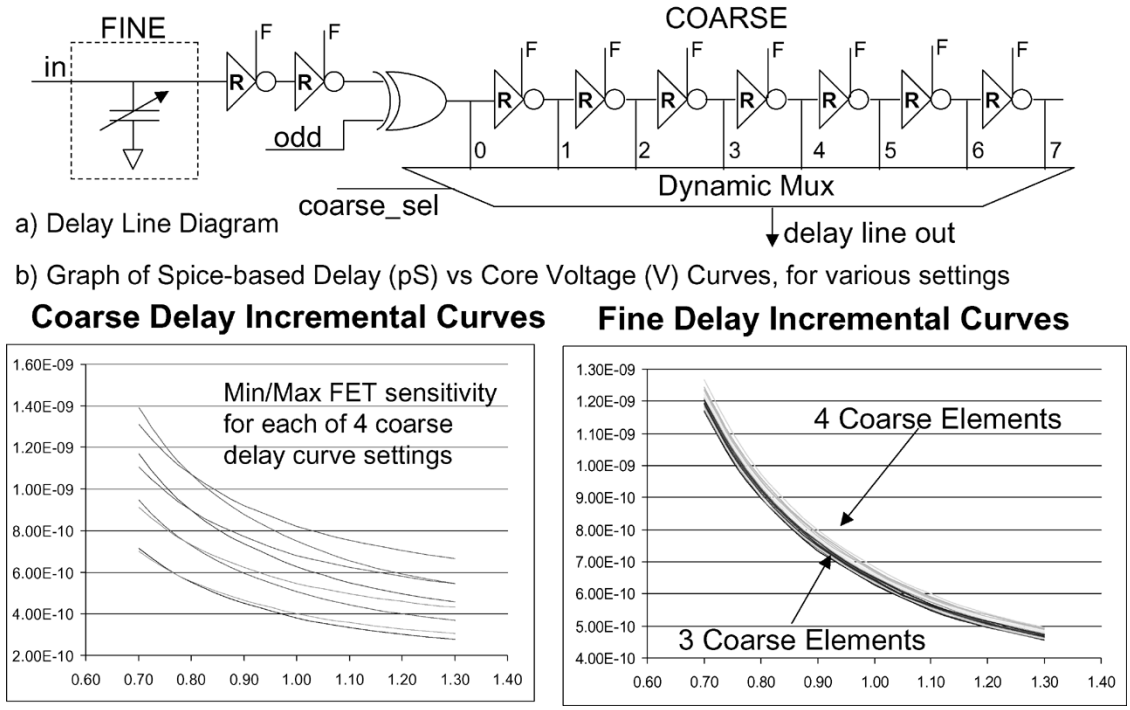


Fig. 6. RVD delay line and SPICE-based voltage-delay curves.

register A while 0B and 1B are set by configuration register B. The RVD FSM sequences the evaluation of the delay lines as shown in Fig. 5(b). Delay lines 0A and 0B propagate the dly0in signal for one clock cycle to perform a supply voltage evaluation while delay lines 1A and 1B are actively cleared. On the next clock cycle delay lines 1A and 1B evaluate while 0A and 0B are cleared. Alternating delay line evaluation and clear cycles in this manner prevents aliasing while achieving one evaluation per cycle bandwidth. This is an $8\times$ bandwidth improvement over the anti-aliasing critical path logic monitor circuit found in [11].

The RVD outputs are generated by two multiplexing phase detector latches and two configurable delay elements. The outputs define core clock period adjustments based on the current operating conditions. "DOWN" indicates timing margin, "HOLD" implies ideal timing conditions and "UP" (DOWN $=$ 0 and HOLD $=$ 0) indicates the need for greater timing margin.

The RVD delay line in Fig. 6 achieves the RVD configurability and matching requirements. The fine delay block in Fig. 6(a) is a configurable capacitor array providing approximately 1.5 ps resolution per bit with approximately 95 ps range under nominal conditions. The inverter labeled "R" is the RVD delay element and is described in more detail in [3].

RVD matching to FET and RC circuits is accomplished by selecting combinations of coarse and fine delay elements. The results of SPICE simulations for various settings are shown in Fig. 6(b). The coarse delay curves show examples of FET and RC limited curves across a range of delay programmings, with delay in ps on the left and voltage on the bottom. Each pair of curves has a FET (high voltage sensitivity) curve with a steep slope and an RC (less voltage-sensitive) curve with a lower slope. The fine delay curves show the granularity of curve matching available by adjusting the fine delay setting for delay curves of the same voltage sensitivity.

IX. VFC Architecture and Implementation

The Montecito VFC consists of voltage-locked loops in which RVD sensor information sets DFD output frequency. Each loop has four RVDs and a single DFD as shown in Fig. 7. To track spatial PVT parameter changes in the Montecito cores, 12 RVDs are distributed around each core. Each major unit has one or two RVDs placed near it to monitor its performance under Foxton supply voltage changes, di/dt events and process and temperature variation. In VFM, core DFD frequency (F_{CORE}) dynamically tracks core voltage via a programmed RVD voltage–frequency $(V$–$F)$ response in the VFC loop. The RVD produces T_{CORE} adjust signals UP, DN (down), and HOLD (implemented via comparator deadzone), and the HOLD capability controls VFC loop stability.

Three DFD zones are used in each core (instead of just one) primarily to reduce frequency adjust latency to supply droops, which in turn reduces performance-wasting frequency guardband which is applied during part binning. As mentioned previously, the use of three distributed DFDs also reduces route distribution matching and signal integrity issues. The three DFDs in each core are located over one phase of flight time from each other, so one extra cycle is needed to adjust adjacent DFDs when one zone first detects a supply droop. The phase compensator state machine (PCSM) filter combines inputs from the four RVDs and the other two DFDs to keep the DFD zones phase aligned within one tick at all times. The PCSM uses a digital proportional-integral control algorithm which incurs a maximum one -tick phase error during adjusts (accounted for as skew in timing analysis) and has no frequency overshoot during increase adjusts.

High VFC bandwidth can track power-managed V_{CORE} modulation as well as high-frequency switching transients. A new

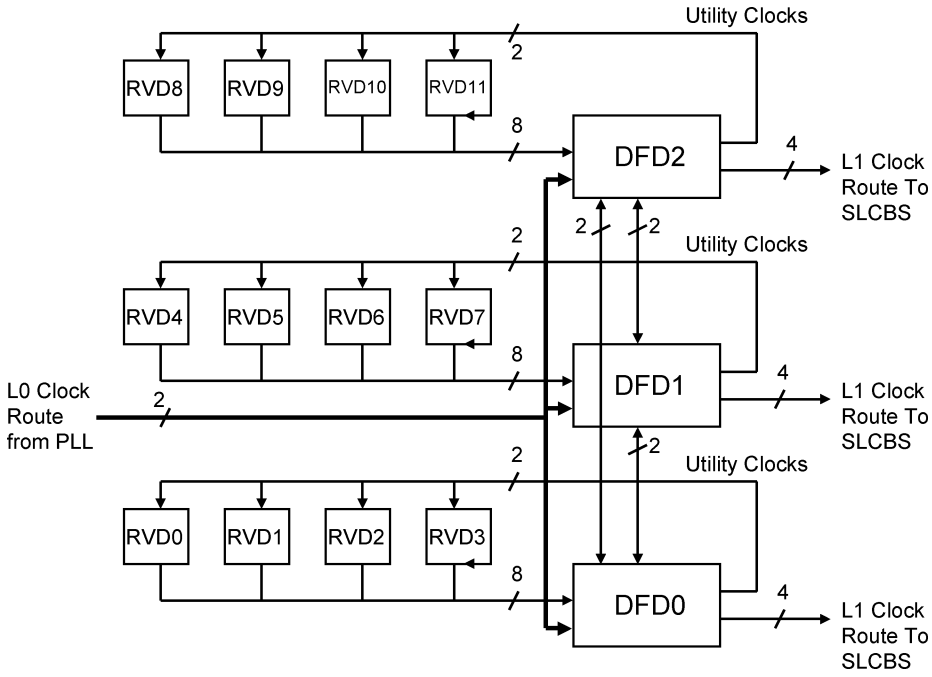


Fig. 7. Montecito voltage–frequency converter system.

frequency is selected with 1.5 cycle average response to a local voltage change event. This frequency change is distributed to latches in ∼700 ps [4]. In each VFC cycle, a DFD utility clock edge: 1) propagates 2400 μm to an RVD; 2) a comparator produces an UP, DN, or HOLD request; 3) routes 2400 μm back to DFD; 4) PCSM arbitrates and resolves comparator metastability; and 5) produces a divisor adjust set up to next clock edge at the DFD. The high bandwidth greatly reduces CPU exposure to voltage transient induced timing issues enabling F_{CORE} to track a voltage transient of up to 30 mV/ns with 700 ps of lead time on average.

In VFM, DFDs synthesize an F_{CORE} range of F_{PLL} to $F_{\mathrm{PLL}}/2$ in 1.6% steps over a V_{CORE} range of 0.8 to 1.2 V. DFDs receive inputs from four local RVDs and from other same-core DFDs. The PCSM arbitrates RVD requests and same-core DFD inputs to derive local DFD divisor adjusts which (a) preserve intra-core DFD phase lock and (b) track programmed V–F response. All DFDs start synchronously in fixed frequency mode using phase 0, and same-core DFD phase lock is maintained in VFM by PCSM arbitration.

Montecito's variable frequency mode is initiated by the Foxton microcontroller. Firmware writes a machine-specific register to send a VFM start signal synchronously to all core DFDs. This enables DFD divisor adjustment in response to RVD requests as described above. The transition into VFM is essentially transparent to Itanium software, and incurs no pipeline flushes or other performance overhead.

When running in fixed frequency mode, core to uncore address and data transfers are synchronous. These interfaces become asynchronous when VFM is initiated. In VFM, the MIB clock continues to run at an integer multiple of the system clock frequency, and the cores run at RVD-determined frequencies. Transfers from the system clock domain into the MIB clock domain are of course synchronous in all modes.

X. Results

Montecito first silicon was received in July 2004 and booted several operating systems within a month using the translation table, DFDs, RAD, and phase alignment. Later, a logical inversion error in the I/O clock—it was incorrectly inverted relative to the cores—was discovered while stopping clocks for scan. The operational effect of this bug had been automatically (and transparently) eliminated by the phase aligner, which simply moved the MIB clock phase an additional 180 degrees during alignment. Other than when stopped, there was no way to see this inadvertent mismatch between the core and I/O clocks. Clock skew has been measured indirectly by scanning out RAD system digital delay codes after the system has stabilized, and typical inter-SLCB skew has been measured at 15 ps. Gater-level clock skew is expected to be within the 20 ps budget allocated for timing paths.

Variable frequency mode was turned on in August 2004, and operating systems were booting and running in VFM by October 2004. Part characterization in VFM has yielded some valuable insights into supply noise and dynamic voltage/power behavior of large multi-core CPUs such as Montecito.

Fig. 8 shows oscilloscope traces of core supply grid and clock system frequency in FFM and VFM during a code-induced droop event. Note that all clock waveforms in Figs. 8–10 are measured using a clock system test port whose frequency is divided by 8 for ease of measurement. For Figs. 8 and 9, the oscilloscope voltage scale is 50 mV/division and time base is 5 ns/division. The supply droop was induced using a self-checking dual core test case, known as the supply droop test case, which maximizes core di/dt, and hence, core supply droop. The first droop magnitude is approximately 160 mV with a maximum slew rate of approximately 60 mV/ns, and the total first droop event duration here is 9 ns (roughly 18 clock cycles).

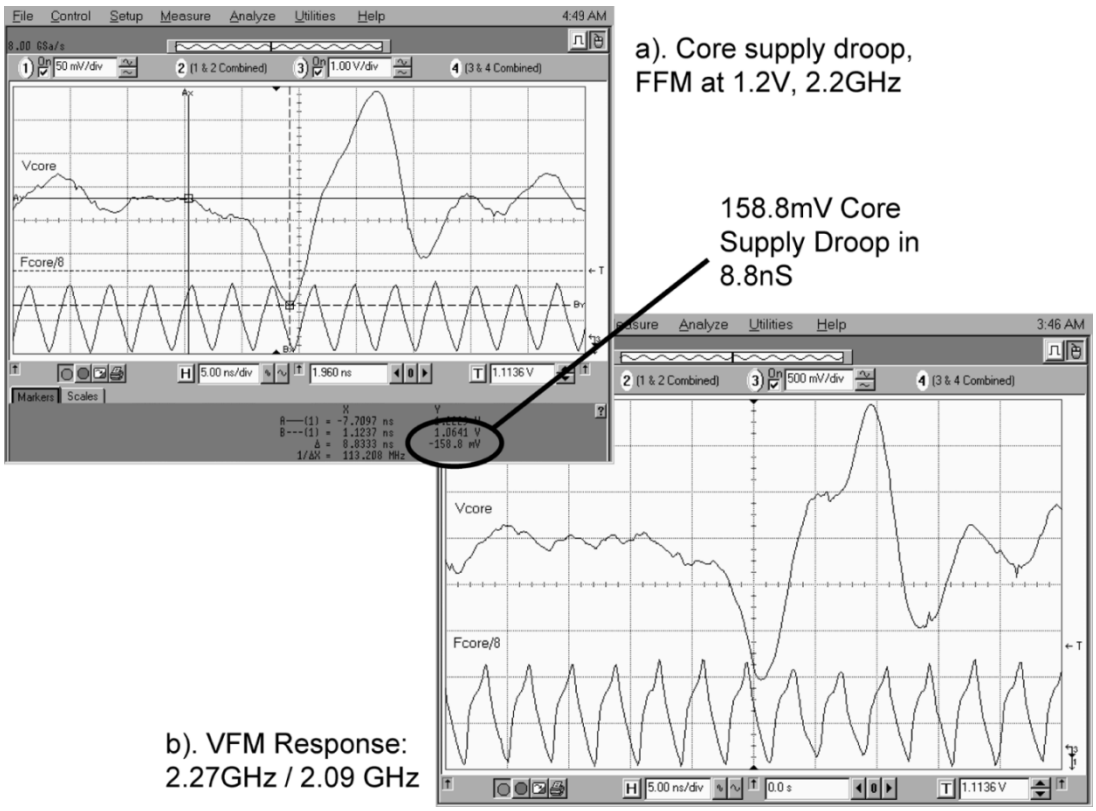


Fig. 8. Oscilloscope traces of core supply droop and clock system response in FFM and VFM.

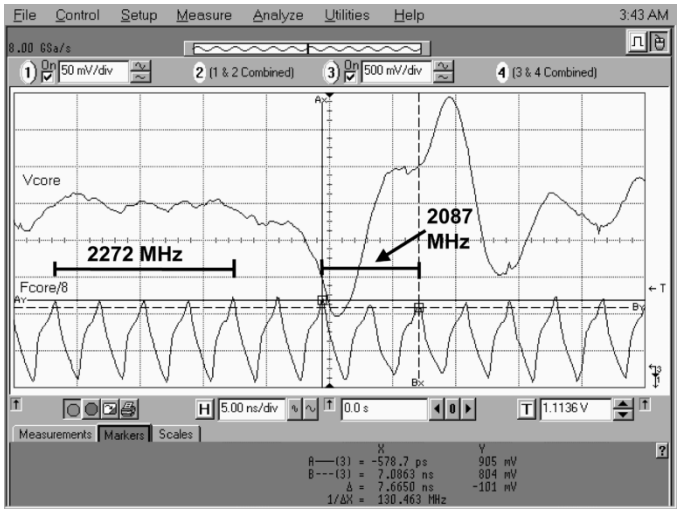


Fig. 9. Oscilloscope trace of clock system VFM response to 150 mV droop at 2.27 GHz, 1.2 V.

As shown in the detail of the VFM droop response in Fig. 9, the average pre-droop frequency over 32 cycles is 2272 MHz, and the frequency averaged over 16 cycles during the event is measured at 2087 MHz. The shift in average frequency of approximately 185 MHz, governed by the programmed RVD curves, confirms that a droop of around 160 mV is seen on-die at the RVDs during this test case. Note that the frequency of the clock system rises back to the pre-droop frequency within 10 ns due to the high bandwidth of the VFC loop. In fact, due to overshoot the frequency actually reaches over 2300 MHz temporarily since critical paths run at higher frequencies at this voltage. The result of the clock system VFC response to supply droop is that code can execute at a higher average frequency under these conditions than is possible in FFM: this is discussed further below. Note that in separate, earlier analyzes, a statically-based on-die supply grid measurement system independently measured an approximate 70 mV droop at 1.2 V and 1.4 GHz while running a different high activity factor test case [3], [10].

Fig. 10 shows the difference between VFM and FFM in their effect on spectral content of the core clocks delivered to Montecito core latches. In Fig. 10(a) in FFM, a single fundamental clock frequency of 1.6 GHz is seen. This is the fixed startup frequency run before entering variable frequency mode. In Fig. 10(b) the system is in VFM. For test purposes a periodic 200 mV peak-peak noise source has been superimposed on the core supply voltage of 1.2 V. This periodic noise source demonstrates VFC adaptation to large-scale voltage adjusts which occur during Foxton power management. Note that the noise source was introduced onto the supply using just a few lines of firmware programmed into the on-die Foxton controller. The major clock frequencies appearing in Fig. 10(b) are a snapshot of core VFCs adjusting to the approximately 2.5 kHz noise source and sampled by the spectrum analyzer. In this test, Itanium code continued to run on the cores uninterrupted during VFM in the presence of the 200 mV noise. These frequency-domain snapshots show the VFCs ability to modulate a high frequency (1.6 GHz) clock with on-die supply noise detected by software-programmed distributed voltage detectors. The fast VFC response shown in Figs. 8–10 allow the chip to run at faster frequencies than in FFM, due to improved

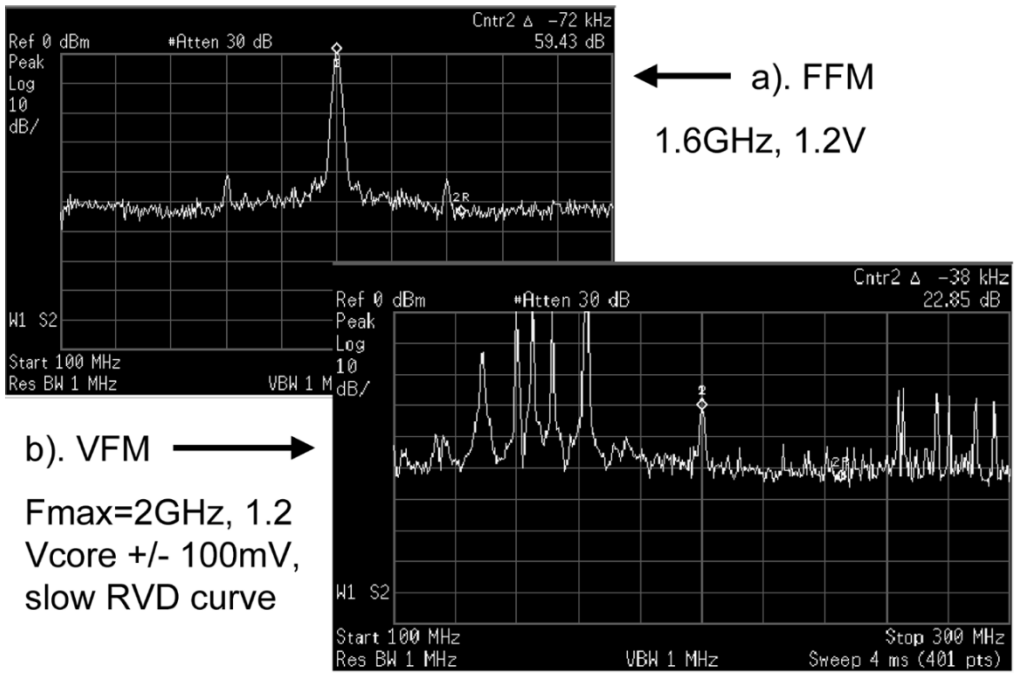


Fig. 10. Core clock FFM/VFM spectral content in presence of 200 mVp-p ∼2.5 kHz supply noise. (a) FFM 1.6 GHz, 1.2 V; (b) VFM Fmax = 2 GHz, 1.2 V_{CORE} ± 100 mV, slow RVD curve.

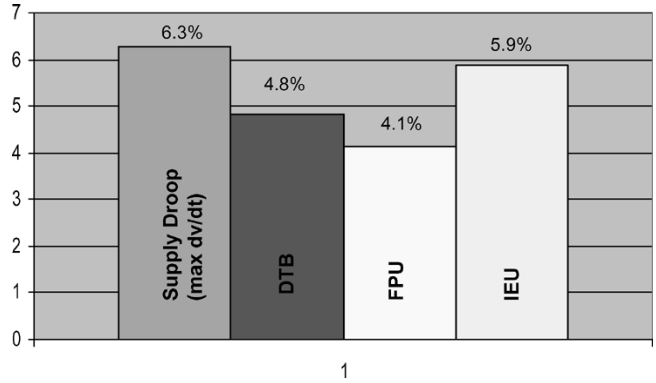


Fig. 11. Measured VFM frequency increase (%) over FFM versus test case.

critical path timing during supply transients. This increases VFM performance over FFM for various test cases as shown in Fig. 11. The DTB and FPU cases are focused critical path tests, whereas the supply droop case is the maximum dv/dt test case detailed in Figs. 8 and 9. VFM frequency is determined by measuring resulting frequency across individual RVD curves and fixed core voltage points—this is known as an RVD shmoo. Average frequencies are reported via on-die frequency counters, and frequencies at which the test case passes are compared to fastest passing FFM shmoo points. Here, the supply droop case passes at average 6% higher frequency in VFM than in FFM due to the adaptive frequency capability. Note that this data was measured at frequencies of approximately 2 GHz at 1.2 V, and the parts were shmooed in FFM at 50 mV/50 MHz granularity.

Clock system results on first silicon included full functionality of FFM, VFM, and all units described above. The clock system has been shown to operate at up to 2.5 GHz at 1.2 V, and enabled first silicon boot of Linux, HPUX, and Windows on multiple platforms with Foxton technology.

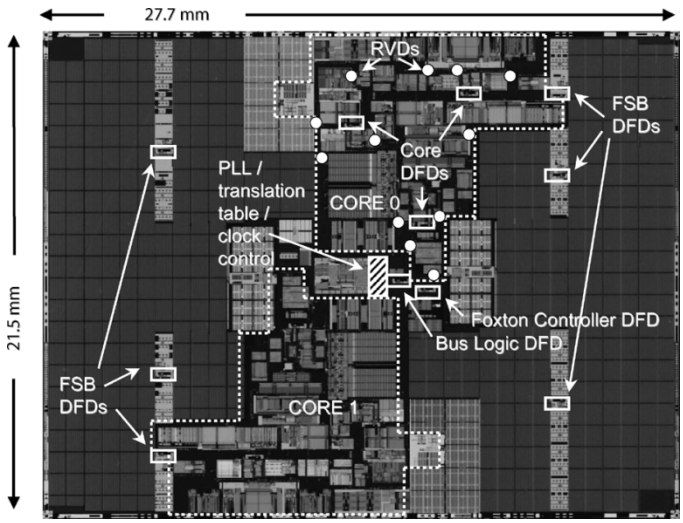


Fig. 12. Die photo.

ACKNOWLEDGMENT

The authors recognize the dedicated efforts of a talented and many-skilled team in designing, verifying, and debugging the Montecito clock system.

REFERENCES

[1] S. Naffziger *et al.*, "The implementation of a 2-core multi-threaded Itanium-family processor," in *IEEE ISSCC Dig. Tech. Papers*, 2005, pp. 182–184.

[2] C. Poirier *et al.*, "Power and temperature control on a 90nm Itanium-family processor," in *IEEE ISSCC Dig. Tech. Papers*, 2005, pp. 304–305.

[3] T. Fischer *et al.*, "A 90nm variable-frequency clock system for a power-managed Itanium-family processor," in *IEEE ISSCC Dig. Tech. Papers*, 2005, pp. 294–296.

[4] E. Fetzer *et al.*, "Clock distribution on a dual-core, multi-threaded Itanium-family processor," in *IEEE ISSCC Dig. Tech. Papers*, 2005, pp. 292–294.

[5] K. Wong *et al.*, "Cascaded PLL design for a 90nm CMOS high performance microprocessor," in *IEEE ISSCC Dig. Tech. Papers*, 2003, pp. 422–504.
[6] S. Naffziger *et al.*, "The implementation of the Itanium 2 microprocessor," *IEEE J. Solid-State Circuits*, vol. 37, no. 11, pp. 1448–1459, Nov. 2002.
[7] S. Sidiropoulos and M. Horowitz, "A semidigital dual delay-locked loop," *IEEE J. Solid-State Circuits*, vol. 32, no. 11, pp. 1683–1692, Nov. 1997.
[8] S. Tam *et al.*, "Clock generation and distribution for the first IA-64 microprocessor," *IEEE J. Solid-State Circuits*, vol. 35, no. 11, pp. 1545–1552, Nov. 2000.
[9] F. Anderson *et al.*, "The core clock system on the next generation Itanium1 microprocessor," in *IEEE ISSCC Dig. Tech. Papers*, 2002, p. 146.
[10] S. Tam *et al.*, "Clock generation and distribution for the 130-nm Itanium2 processor with 6-MB on-die L3 cache," *IEEE J. Solid-State Circuits*, vol. 39, no. 4, pp. 636–642, Apr. 2004.
[11] E. Alon *et al.*, "Circuits and techniques for high-resolution measurement of on-chip power supply noise," in *IEEE Symp. VLSI Circuits*, Jun. 2004, pp. 102–105.
[12] T. Kuroda *et al.*, "Variable supply-voltage scheme for low-power high-speed CMOS digital design," *IEEE J. Solid-State Circuits*, vol. 33, no. 3, pp. 454–462, Mar. 1998.
[13] C. Lichtenau *et al.*, "PowerTune: advanced frequency and power scaling on PowerPC 970+," in *IEEE ISSCC Dig. Tech. Papers*, 2004, pp. 356–357.

Tim Fischer received the M.S. degree in computer engineering from the University of Cincinnati, Cincinnati, OH, in 1989.

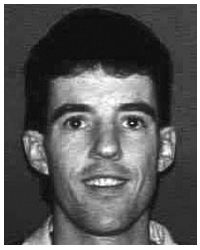
He worked for Digital Equipment Corporation from 1989 to 1998 designing Alpha and VAX microprocessors, including the EV5 floating point and EV6 instruction issue units. He joined Hewlett-Packard in 1998 as a technical contributor working on L1 cache design and global circuit methodology for Itanium 2. On the Montecito processor he worked on global circuit methodology, power reduction, and clock system architecture, design, and verification.

Jayen Desai received the B.S. and M.S. degrees in electrical engineering from Stanford University, Stanford, CA, in 1994.

From 1995 to 2005, he worked at Hewlett-Packard, Fort Collins, CO, on several PA-RISC and Itanium2 processors in the areas of timing analysis, clock design, synchronizer design, custom digital design, and I/O design. He is now a staff engineer at Intel's Fort Collins Design Center and is working on future Itanium2 processors.

Bruce Doyle (M'84) received the B.S.E.E degree from Carleton University, Ottawa, ON, Canada, in 1984.

Since 1984, he has been involved in the design of memories, graphics processors, analog circuits, and mixed-signal ICs. He joined Hewlett-Packard, Fort Collins, CO, in 2002 working on the Itanium line of microprocessors as a physical designer on high-speed I/Os, the clock system and the thermal system. He joined Intel in February 2005 at the Fort Collins Design Center. He is currently technical lead for the clock system on the latest generation Itanium processor. His expertise includes high-speed circuit design, analog circuit design, and mixed-signal design verification. He holds 11 U.S. patents.



Samuel Naffziger (M'02) received the B.S. degree in electrical engineering from the California Institute of Technology, Pasadena, in 1988, and the M.S.E.E. degree from Stanford University, Stanford, CA, in 1993.

He joined Hewlett Packard in 1988, and spent eight years working on various aspects of the PA-RISC processor development including floating point out-of-order execution and circuit methodologies. He then became part of the Itanium2 Joint Development Team with Intel Corporation, Fort Collins, CO, and has led the design of both the first Itanium2 processor, and most recently, the Montecito design. He is currently the Director of Itanium Circuits and Technology within Intel. He holds 59 U.S. patents on processor circuits and architecture.

Mr. Naffziger chairs the International Solid-State Circuits Conference Digital Subcommittee, and is an Intel Fellow.

Ben Patella (M'04) received the B.S. degree in computer engineering and the M.S.E.E. degree from the University of Colorado, Boulder, CO.

He is a Design Engineer at Intel Corporation, Fort Collins, CO, and works on the Itanium family of processors in the Digital Enterprise Group. His main focus has been circuit and architectural contributions to Montecito's voltage to frequency converter, which enables variable frequency clocks for Foxton power management. Patella joined Intel in 2005 from the Hewlett-Packard processor design team. He joined Hewlett-Packard in 2001.

Mr. Patella's thesis, "Implementation of a Low-Power, High Frequency Digital Pulse Width Modulation Controller," won the Engineering College Award for Excellence in Scholarship and Research.