

# Sub-Clock Power-Gating Technique for Minimising Leakage Power During Active Mode

Jatin N. Mistry\*, Bashir M. Al-Hashimi\*, David Flynn<sup>†</sup> and Stephen Hill<sup>†</sup>

\*School of Electronics & Computer Science, University of Southampton, U.K. {jnm106, bmah}@ecs.soton.ac.uk

<sup>†</sup>ARM Ltd., Cambridge, U.K. {David.Flynn, Stephen.Hill}@arm.com

**Abstract**—This paper presents a new technique, called sub-clock power gating, for reducing leakage power in digital circuits. The proposed technique works concurrently with voltage and frequency scaling and power reduction is achieved by power gating within the clock cycle during active mode unlike traditional power gating which is applied during idle mode. The proposed technique can be implemented using standard EDA tools with simple modifications to the standard power gating design flow. Using a 90nm technology library, the technique is validated using two case studies: 16-bit parallel multiplier and ARM Cortex-M0<sup>™</sup> microprocessor, provided by our industrial project partner. Compared to designs without sub-clock power gating, in a given power budget, we show that leakage power saved allows 45x and 2.5x improvements in energy efficiency in the case of multiplier and microprocessor, respectively.

## I. INTRODUCTION

Technology scaling has been the driving force behind the growth of the mobile computing industry. Each new technology node benefits from increased performance, increased integration and reduced dynamic power through supply voltage scaling [1]. An undesirable side-effect of technology scaling, however, is the significant increase in leakage current. The reduction in supply voltage necessitates the reduction of threshold voltage to maintain performance [1], which exponentially increases sub-threshold leakage. Furthermore, technology scaling has reduced transistor gate oxide thickness to only a few atoms, resulting in a significant rise in gate leakage current [2]. At sub-65nm technology nodes, leakage power is considered to be as dominant as dynamic power [2]. When logic is in idle mode, i.e. when it is not doing useful work, this leakage current becomes an unnecessarily significant drain of power within the circuit.

A number of solutions have been proposed for reducing the leakage power dissipation of digital circuits including the use of dual-threshold logic [3], exploitation of transistor stacks [4] and power gating [5]. Power gating shuts down parts of a digital circuit and is effective at reducing leakage power during idle mode; it has been reported to reduce leakage power by up to 25x in the ARM926EJ [5]. The power gated circuit is connected to the supply rail through a high- $V_t$  PMOS sleep transistor, disabling the high- $V_t$  transistor during extended idle periods shuts down the power gated circuit limiting leakage power dissipation. Any state that needs to be retained is achieved with state retention registers and

the activation/deactivation of power is controlled by a power gating controller.

Leakage, however, can be a significant drain of power during active mode, i.e. when the digital circuit is doing useful work. Voltage and frequency scaling have proved to be an effective technique to reduce active mode power for power constrained digital circuits [5]. However, application of aggressive frequency scaling to below maximum frequency,  $F_{max}$ , increases idle time of combinational logic within the clock cycle dissipating unnecessary leakage power. The leakage power dissipated requires the circuit to operate at a lower clock frequency to meet the given power budget, as total power is made up of dynamic and leakage power, consequently making the digital circuit less energy efficient (Energy per operation = Power x Time). The parasitic leakage dissipation can be a significant problem for applications such as a wireless sensor node powered by an energy harvester. Such systems, often not performance critical, demand a strict power budget to meet harvester's output power and high energy efficiency to optimize use of harvested-energy [6].

This paper presents an alternative approach of power gating that works concurrently with voltage and frequency scaling. It capitalises on the idle time of combinational logic in the clock cycle during active mode, to reduce overall power consumption by power gating within the clock cycle. The proposed technique, called sub-clock power gating (SCPG), separates the design into an always-on sequential logic domain and a power gated combinational logic domain. It therefore does not need the retention registers required by traditional power gating. The power gates are controlled by the toggling of the clock, also alleviating the need for a power gating controller, reducing implementation complexity. By working in synergy with voltage and frequency scaling, both dynamic and leakage power are simultaneously reduced and an SCPG design can meet a given power budget at a higher operating frequency, thus maximising energy efficiency. Recent work has reported the application of power gating in active mode and operates by turning off unused logic in a similar manner to clock-gating by utilising clock enable signals [7] but targets leakage reduction of logic on a per clock cycle basis. The work proposed in this paper, however, is the first study that demonstrates the application of power gating to reduce leakage power of logic within the clock cycle. The rest of this paper is organized as follows. Section II describes how the proposed technique can be used concurrently with voltage and

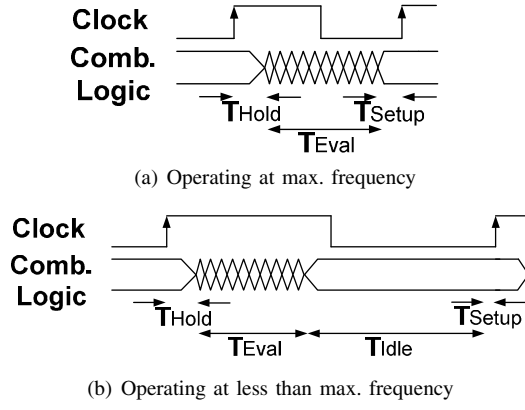


Fig. 1. Idle time within clock cycle from reducing clock frequency

frequency scaling to reduce leakage power dissipation. Section III validates the proposed sub-clock power gating technique using two test cases: 16-bit multiplier and an ARM Cortex-M0 microprocessor. Section IV compares the proposed technique with the sub-threshold technique [8]. Section V concludes the paper.

## II. PROPOSED TECHNIQUE

Voltage and frequency scaling is an effective way of reducing dynamic power during active mode for a power constrained digital circuit [5]. The scaling of supply voltage ( $V_{DD}$ ) gives significant reduction in power due to its quadratic relation with dynamic power, but it is well known that technical challenges and stability issues arise when lowering the supply voltage to near or below the threshold voltage [8]. Operating at maximum frequency ( $F_{max}$ ) at a reduced supply voltage may still exceed a given power budget. For example, an ARM Cortex-M0 targeted at an energy harvesting application consumes 0.9mW with a supply voltage of 0.6V using a 90nm technology library, but a typical energy harvester power budget is between tens and hundreds of  $\mu$ Ws [6]. Frequency scaling can provide further (linear) reduction in dynamic power to meet a given power budget. Overall power is reduced with reduction in frequency but the clock period ( $T_{clk}$ ) becomes longer than the combined hold time ( $T_{hold}$ ), evaluation time of the combinational logic ( $T_{eval}$ ) and setup time ( $T_{setup}$ ) resulting in idle time ( $T_{idle}$ ) within the clock cycle, as demonstrated in Fig. 1. With leakage power a major concern in nanometer designs, this idle time can represent a significant drain of power during active mode. The total power is made up of a sum of dynamic power and the leakage power and therefore requires the circuit to operate at a lower clock frequency to meet a given power budget, than if power was not lost to leakage. This consequently consumes more energy per operation as energy is the product of power and time reducing energy efficiency. The proposed sub-clock power gating (SCPG) technique capitalises on this idle time of combinational logic to reduce leakage power by power gating within the clock cycle during active mode. By working concurrently with voltage and frequency scaling to reduce dynamic and leakage power simultaneously, the same power budget can be achieved at a higher operating frequency,

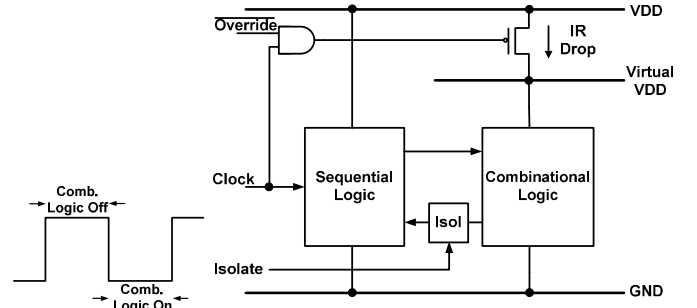


Fig. 2. Sub-Clock Power Gating Technique

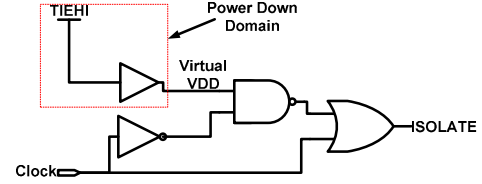


Fig. 3. Isolation Circuit

and subsequently means the same digital circuit operates more efficiently from an energy point of view.

The architecture of a circuit with the proposed sub-clock power gating is shown in Fig. 2. The combinational logic of the circuit is connected to the supply rail through a high- $V_t$  PMOS header transistor whereas the sequential logic is connected directly to the supply rail. This consequently means that, during operation, only the combinational logic can be powered down. In traditional power gating the registers in the circuit would also be power gated requiring retention registers for saving and restoring state [5]. As the registers in a sub-clock based architecture are always-on retention registers are not needed.

The PMOS header transistor used to power gate the combinational region is controlled by the clock signal  $ANDed$  with an active low override signal. The active low override signal allows deactivation of the power gating by forcing the power gate on continuously. As the PMOS header transistor is controlled by the clock, the combinational logic is power gated during the clock's active phase and is functional when the clock is low, Fig. 2. The leakage power saving is therefore influenced by two variables: the operational clock frequency and the duty cycle of the clock. As the clock frequency is reduced, the idle time of the logic ( $T_{idle}$ ) becomes greater as the difference between the evaluation time ( $T_{eval}$ ) and the clock period ( $T_{clk}$ ) increases, Fig. 1, presenting greater potential leakage power saving. Using a 50% duty cycle saves leakage power for half the clock period ( $T_{clk}/2$ ) but restricts the application of SCPG to when  $T_{eval} < T_{clk}/2$  to allow enough time for evaluation of the combinational logic. Changing the duty cycle, on the other hand, allows the application of SCPG even when  $T_{clk}/2 < T_{eval} < T_{clk}$  by decreasing the duty cycle. Also, when  $T_{eval} \ll T_{clk}$ , changing the duty cycle to a greater value capitalises on all the logic's idle time to provide maximum leakage power saving, as demonstrated (section III).

A similarity between traditional power gating and the pro-

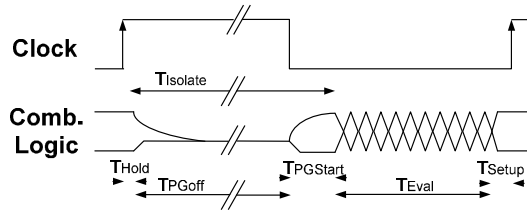


Fig. 4. Sub-Clock Power Gating Timing

posed SCPG technique is the need for isolation gates [5]. Isolation is inserted on all outputs from the power down domain and ensures they do not cause corruption or short circuit currents in the always-on regions, when the power is gated to the region, by clamping the output signals (marked as Isol in Fig. 2). In traditional power gating, power gates, isolation cells and retention registers are controlled with a power gating controller state machine. During power down, first the clocks are stopped, the output signals are clamped, the state is retained and then the power is switched off; the process is reversed for power-up. Furthermore, extra routing of these control signals is required, introducing area and power overhead. In the proposed SCPG technique, the power gates are controlled by the clock and the registers are always-on, and therefore a power gating controller is not needed saving area and power and reducing implementation complexity. Routing of the control signals is also minimised as the extensive, high-fanout clock tree of a processor can be exploited for the power gating control signal. However, the nature of SCPG raises a problem with controlling the isolation control signal (ISOLATE, Fig. 2). As the power gating is controlled within the clock cycle, a state machine cannot be used to time the clamping of the output signals. Instead a simple circuit is introduced to provide the isolation control signal and is adaptive to the behaviour of the virtual supply rail, Fig. 3. The circuit uses the clock and the value of the virtual supply rail (VDDV), by connecting to a TIEHI in the power down domain, as primary inputs. By doing this the isolation can be activated as soon as the clock goes high but is delayed in deactivation until the virtual supply rail represents a logic '1'. The complete timing diagram of a sub-clock power gating design is given in Fig. 4. The power is switched off at the positive edge of the clock, but the delay in the collapse of the virtual rail and activation of isolation maintains the hold time ( $T_{hold}$ ) required for propagating the state to the registers. The combinational domain is then power gated for the remainder of the active part of the clock ( $T_{PGOff}$ ) reducing leakage power dissipation. The power is restored at the negative edge of the clock but isolation is held until the combinational logic is active again ( $T_{PGStart}$ ) ensuring problems with short circuit currents do not arise. The combinational logic then evaluates the next state ( $T_{eval}$ ) in the available time, meeting setup time ( $T_{setup}$ ), before the process repeats at the next positive edge.

The design flow to implement a circuit including the proposed SCPG is shown in Fig. 5; two additional steps are added to a traditional design flow and are indicated. Step one requires the combinational and sequential logic to be separated to apply the proposed SCPG technique. This allows them to be assigned

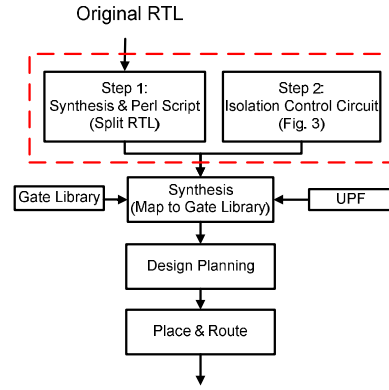


Fig. 5. Design flow of the proposed sub-clock power gating technique

to separate power domains later in the implementation flow. This is achieved by parsing the netlist of a design and moving the combinational logic to a separate verilog module. Step 2 requires the custom isolation circuitry presented in Fig. 3 to be combined with the new split netlist before the entire design is synthesized. The additional steps presented here are fully compatible with a traditional power gating design flow using a UPF (Unified Power Format) file to define the power gating strategy. During 'Design Planning', however, it is recommended that the combinational logic domain is located in the center of the design to alleviate problems with routing congestion between the combinational logic and the sequential logic domains. The remainder of the implementation flow - placement, clock tree synthesis & routing - is identical to a traditional power gating implementation flow.

### III. EXPERIMENTAL RESULTS

To validate the sub-clock power gating technique, two case studies were used: a 16-bit parallel binary multiplier and an ARM Cortex M0 microprocessor. The designs are both implemented using the flow described in section II, Fig. 5, using a 90nm technology library<sup>1</sup> and the Synopsys EDA tool suite. The post place and route spice netlist is simulated at a supply voltage of 0.6V using Synopsys HSpice and the power and energy values are recorded, Table I and II.

An integral part of implementing power gating is the choice of sleep transistors. The inclusion of the header transistors introduces a small IR drop (Fig. 2). As reported in previous publications, the header transistor size, the number of headers and their arrangement directly affects the IR drop across the power domain [9], [5]. With a lower IR drop the impact in performance is reduced and the time taken to reach an active state from power down is also reduced. However, including many header transistors can have a negative impact on ground bounce and in-rush current [5]. The 90nm process used for the two test cases has a range of power gating transistor sizes. In our experimentation we have investigated the affects of header transistor sizing on performance. It has been found from synthesis and simulation that the best IR drop can be achieved with X2 size transistors for the 16-bit multiplier, and X4 size transistors for the Cortex-M0.

<sup>1</sup>Synopsys 90nm Education Kit available from Synopsys

TABLE I  
POWER AND ENERGY PER OPERATION OF SUB-CLOCK POWER GATED  
MULTIPLIER, VDD=0.6V

| Clock<br>(MHz) | No Power Gating |                | Proposed SCPG |                |               | Proposed SCPG-Max |                |               |
|----------------|-----------------|----------------|---------------|----------------|---------------|-------------------|----------------|---------------|
|                | Power<br>(uW)   | Energy<br>(pJ) | Power<br>(uW) | Energy<br>(pJ) | Saving<br>(%) | Power<br>(uW)     | Energy<br>(pJ) | Saving<br>(%) |
| 0.01           | 29.23           | 2923           | 17.58         | 1758           | 39.9          | 5.80              | 580.2          | 80.2          |
| 0.1            | 29.44           | 294.4          | 18.02         | 180.2          | 38.8          | 6.33              | 63.25          | 78.5          |
| 1              | 31.54           | 31.54          | 22.38         | 22.38          | 29.0          | 11.55             | 11.55          | 63.4          |
| 2              | 33.87           | 16.94          | 27.05         | 13.53          | 20.1          | 17.35             | 8.68           | 48.8          |
| 5              | 40.88           | 8.18           | 37.16         | 7.43           | 9.1           | 32.78             | 6.56           | 19.8          |
| 8              | 47.89           | 5.99           | 44.84         | 5.61           | 6.4           | 43.45             | 5.43           | 9.3           |
| 10             | 52.62           | 5.26           | 49.89         | 4.99           | 5.2           | 49.06             | 4.91           | 6.8           |
| 14.3           | 62.67           | 4.38           | 60.61         | 4.24           | 3.3           | 60.59             | 4.24           | 3.3           |

#### A. Case Study 1: 16-bit Multiplier

This circuit was chosen because of its large concentration of combinational logic to highlight gains in large datapath blocks. The inclusion of the SCPG technique introduces approximately 3.9% increase in area. This can be accounted to the power gating circuitry and the addition of buffers to compensate for the splitting of the combinational and sequential logic into separate power domains. Table I gives the average power dissipation and energy per operation values for a range of clock frequencies with no SCPG, SCPG using 50% clock duty cycle, and using greater than 50% duty cycle to maximise leakage power reduction. It can be seen the proposed SCPG technique reduces the average power when compared to no SCPG, and greater savings are achievable at lower frequencies because of increased idle time of logic. Furthermore, changing the duty cycle of the clock to a higher value (Proposed SCPG-Max) maximises the savings achievable. For example, at 10kHz, the saving rises from 39.9% to 80.2%.

Fig. 6(a) depicts the trends in average power dissipation with increasing clock frequency. As can be seen the average power dissipation of the 3 setups converge with increase in clock frequency because, as the idle time of logic decreases, the power dissipated from switching the header transistor begins to dominate over leakage power savings. Thus, the point is reached when the power dissipated switching the header equals the leakage power saved and beyond that frequency an SCPG design would not save any power. For the multiplier, it was found that the 3 setups converge at approximately 15MHz and beyond that frequency the circuit was unable to save power using SCPG compared to the original design. Fig. 6(b) shows the energy per operation against clock frequency for the three designs. As expected, energy per operation decreases with increasing clock frequency but note that at a given frequency the SCPG design is more energy efficient due to savings of leakage power. The leakage power saving also means, given an average power budget, a higher frequency can be achieved with an SCPG design. The utility of this is illustrated with an example. One target application envisaged for the proposed technique is designs with tight power budgets, e.g., a wireless sensor node powered by an energy harvester. Given a typical energy harvester power budget of  $30\mu\text{W}$  [6], the multiplier with no SCPG would need to operate at 100kHz and would consume 294.4pJ/operation (Table I). With SCPG,

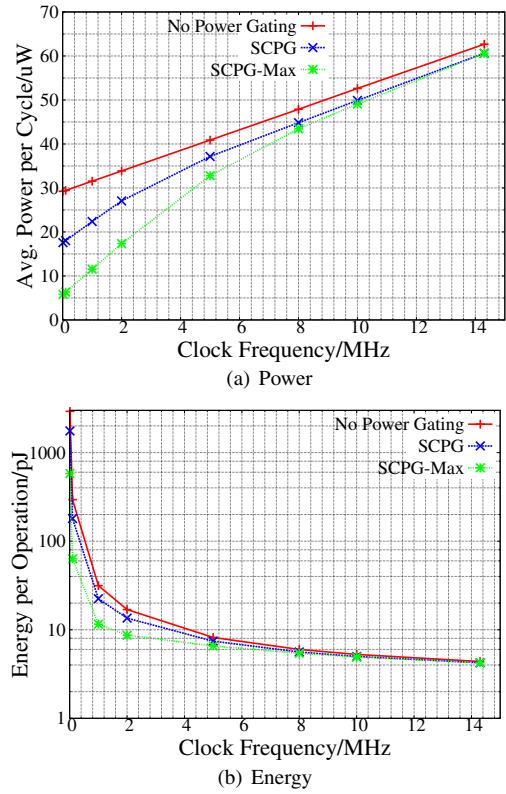


Fig. 6. 16-bit binary multiplier circuit, VDD=0.6V

the given power budget can be met at an operating frequency of approximately 2MHz consuming 13.33pJ/operation (Table I). Furthermore, if the clock duty cycle is increased, SCPG can achieve an operating frequency of approximately 5MHz consuming 6.56pJ/operation (Table I), implying a 50x increase in clock frequency with 45x improvement in energy efficiency within the same power budget.

#### B. Case Study 2: ARM Cortex M0

The ARM Cortex M0 was chosen because of its ultra-low power design, which serves as a good candidate to demonstrate the gains of SCPG in a real world application setting. The microprocessor has 3 stage pipeline and a 32-bit, RISC architecture with the compact Thumb2™ instruction set. As the microprocessor provided by our industrial partner is an RTL core, it allows the SCPG technique to be tested using the design flow presented in Section II (Fig. 5). The inclusion of the SCPG technique increases area by approximately 6.6% due to the additional circuitry.

To obtain the power characteristics of the microprocessor at a specific performance point, the Dhrystone benchmark was used as it represents a range of application workloads [10]. To keep HSpice simulation time reasonable, the steps presented next were followed. The Cortex-M0 netlist was simulated with the Dhrystone benchmark in Mentor Modelsim and a value change dump (VCD) file was created from the switching activity of the circuit. The complete benchmark (3700 vectors) was divided into groups of 10 vectors and each groups' average switching activity was obtained with Synopsys Primetime-PX using the VCD file obtained (Fig. 7).

TABLE II  
POWER AND ENERGY PER OPERATION OF SUB-CLOCK POWER GATED  
CORTEX-M0, VDD=0.6V

| Clock<br>(MHz) | No Power Gating |                | Proposed SCPG |                |               | Proposed SCPG-Max |                |               |
|----------------|-----------------|----------------|---------------|----------------|---------------|-------------------|----------------|---------------|
|                | Power<br>(uW)   | Energy<br>(pJ) | Power<br>(uW) | Energy<br>(pJ) | Saving<br>(%) | Power<br>(uW)     | Energy<br>(pJ) | Saving<br>(%) |
| 0.01           | 243.65          | 24364          | 175.19        | 17518          | 28.1          | 104.56            | 10456          | 57.1          |
| 0.1            | 244.59          | 2445.9         | 179.37        | 1793.6         | 26.7          | 109.31            | 1093           | 55.3          |
| 1              | 253.92          | 253.92         | 220.87        | 220.87         | 13.0          | 157.08            | 157            | 38.1          |
| 2              | 264.29          | 132.14         | 260.87        | 130.48         | 1.3           | 209.43            | 105            | 20.8          |
| 5              | 295.43          | 59.09          | 303.21        | 60.64          | -2.7          | 289.79            | 57.96          | 1.9           |
| 10             | 347.30          | 34.73          | 388.63        | 38.86          | -12           | 387.52            | 38.75          | -11           |

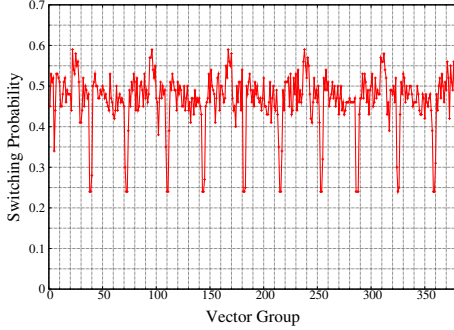


Fig. 7. Switching probability of the Cortex M0 for each set of 10 vectors from Dhrystone benchmark

Three groups of test vectors representing maximum, minimum and average switching activity were then extracted from these 370 groups. The test vectors representing these three test cases were then Simulated in HSpice and the power numbers obtained were used to estimate the average power dissipation from a complete Dhrystone benchmark.

The simulation results for the Cortex-M0 design are reported in the same format as those for the 16-bit multiplier in Section III-A, Table I. Although savings are achieved for a range of frequencies, e.g., 28.1% and 57.1% at 10 kHz (Table II, 1st row), it can be seen that lower savings are achieved compared to the multiplier at a given clock frequency (28.1% vs 39.9% at 10 kHz, Table I). This difference can be explained by the contrast in size between the two designs. The increased concentration of combinational logic in the Cortex-M0 (6747 gates vs 556 gates in the multiplier) increases the energy required to charge the virtual supply rail. Furthermore, as the power domain is restored back to an active state the crowbar currents within the power gated region are more significant in a larger design. These two effects therefore increase the power overhead of power gating in a larger design. Also, this increased power overhead means a lower convergence point as can be seen in Fig. 8(a). Note that the Cortex M0 designs converge around 5MHz, whereas the multiplier designs converge at approximately 15MHz (Fig. 6(a)).

Despite the increased power overhead, the leakage power reduction achievable with SCPG allows higher energy efficiency to be achieved (Fig. 8(b)), and this is demonstrated with another example. Given a typical energy harvester power budget of  $250\mu\text{W}$  [6], a Cortex-M0 design without SCPG would need to operate at approximately 1MHz consuming 253pJ/operation (Table II). On the other hand, an SCPG design using 50% clock duty cycle operates at approximately 2MHz

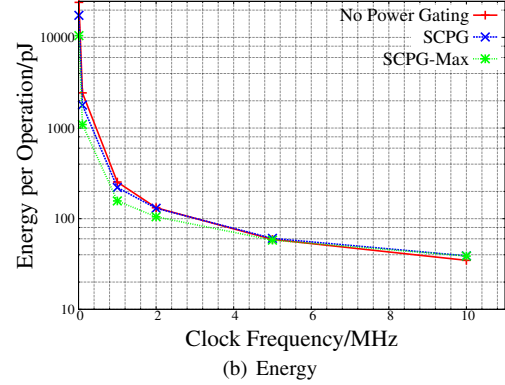
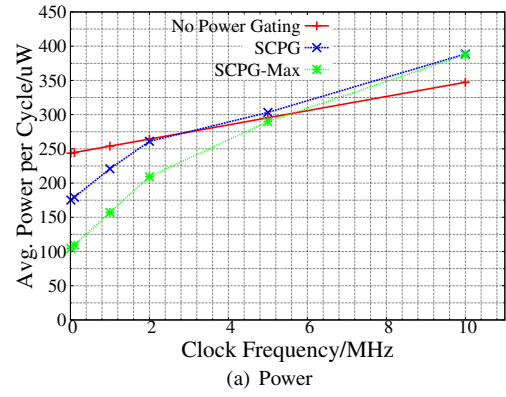


Fig. 8. Cortex M0, VDD=0.6V

consuming 130.48pJ/operation, while operating at maximum duty cycle consumes  $< 105\text{pJ/operation}$  at an operating frequency between 2-5MHz (Table II). This represents over 2.5x improvement in energy efficiency achieved by operating at over 2x higher clock frequency.

#### IV. COMPARATIVE ANALYSIS WITH SUB-THRESHOLD

Sub-threshold design technique enables realization of minimum energy computation [8]. In sub-threshold operation, the supply voltage is lowered beyond the threshold voltage, until a minimum energy point is found where dynamic energy equals leakage energy. The work proposed in this paper can be considered as an alternate to sub-threshold technique to achieve low power operation whilst maximising energy efficiency. This section investigates the sub-threshold operation of the two test circuits with the aim of establishing the performance of sub-clock power gating relative to the sub-threshold technique.

Fig. 9 shows the energy per operation against supply voltage of the 16-bit multiplier when using sub-threshold design. It was found from HSpice simulation that the minimum energy point (1.7pJ/operation) is obtained at a supply voltage of 310mV corresponding to an operating frequency of approximately 10MHz. For comparison with an SCPG design, the average power of  $17\mu\text{W}$  at this point is set as the power budget. From Table I, it can be seen that this dictates operation at 2MHz consuming 8.68pJ/operation; a 5x reduction in performance and a 5x increase in energy. This shows, as expected, that a sub-threshold technique offers better energy efficiency than SCPG since sub-threshold operation provides minimum energy computation. Note that if the power budget is

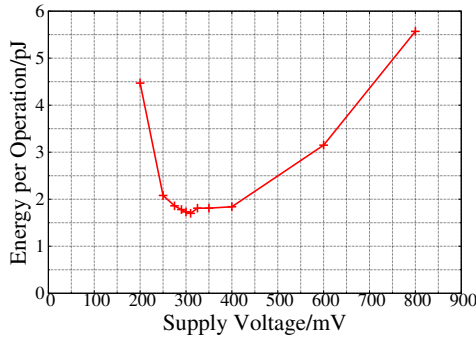


Fig. 9. Supply voltage Vs Energy per operation, 16-bit binary multiplier

increased, the difference between the two approaches narrows. For instance, setting a power budget of  $40\mu\text{W}$  results in a difference in energy of 2.9x.

Fig. 10 shows the energy per operation at different supply voltages for the Cortex-M0 using sub-threshold design. The same trend is observed here as in case of 16-bit multiplier. Note, however, that the increased density of logic in this circuit pushes the minimum energy point towards a higher supply voltage. This is because the leakage energy of the increased number of gates dominates at a higher clock frequency. Simulation locates the minimum energy point at a supply voltage of 450mV, corresponding to an operating frequency of 24MHz, consuming 12.01pJ/operation or average power consumption of  $288.24\mu\text{W}$ . At similar power budget, the sub-clock power gated Cortex-M0 (Table II) has 5x reduction in performance and a 4.8x increase in energy.

Although sub-threshold technique offers better energy efficiency than sub-clock power gating, it is limited to applications where only low performance is needed due to the large delay introduced by operating at an ultra-low voltage [8]. SCPG on the other hand provides a performance/power trade-off. Assuming a larger power budget is available, increasing the clock frequency allows high performance to be achieved whilst still minimising leakage power. Additionally, the 'Override' signal (Fig. 2) enables the system to peak to maximum performance, allowing the digital circuit to toggle between low power, low performance (kHzs) and high power, high performance (MHzs) states, unlike subthreshold which is limited to slow operation. The TI MSP430, commonly used in wireless sensor networks, is a prime example of how the performance/power trade-off can be used. The microcontroller uses a slow clock for background tasks whereas a fast clock is required for signal processing. A digital circuit designed for sub-threshold technique also introduces technical challenges due to the ultra-low operating voltages. The circuit is more sensitive to process variations such as variations in threshold voltage and temperature [8]. The increased sensitivity can skew the minimum energy point significantly making the designing of a sub-threshold circuit much harder [8]. In comparison, SCPG operates above threshold voltage maintaining greater stability with process and temperature variations.

## V. CONCLUSION

This paper presented the first study into the application of power gating within the clock cycle to reduce leakage

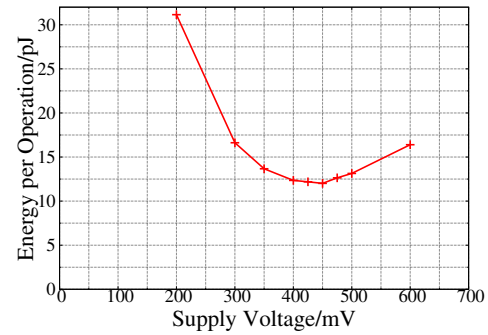


Fig. 10. Supply voltage Vs Energy per operation, Cortex-M0

power and works concurrently with voltage and frequency scaling. It is shown that it is possible to reduce leakage power of combinational logic during active mode using the proposed sub-clock power gating technique. Since frequency scaling below maximum frequency results in idle time of combinational logic within the clock cycle, sub-clock power gating capitalises on this idle time of combinational logic to achieve power saving. The proposed sub-clock power gating technique has been validated with a 16-bit multiplier and an ARM Cortex-M0 microprocessor using a 90nm technology library. It has been shown that the proposed technique offers considerable saving in leakage power allowing the digital circuit to operate at a higher clock frequency and consequently more energy efficiently within the same given power budget. The proposed technique is also compatible with standard EDA tools.

## ACKNOWLEDGMENT

The authors wish to thank Mustafa Imran Ali & James Myers for valuable discussions & EPSRC-UK for funding this work, grant number EP/G067740/1.

## REFERENCES

- [1] S. Borkar, "Design Challenges of Technology Scaling," *IEEE Micro*, vol. 19, pp. 23–29, 1999.
- [2] A. Agarwal, S. Mukhopadhyay, A. Raychowdhury, K. Roy, and C. Kim, "Leakage Power Analysis and Reduction for Nanoscale Circuits," *IEEE Micro*, vol. 26, 2006.
- [3] L. Wei, Z. Chen, K. Roy, M. Johnson, Y. Ye, and V. De, "Design and Optimization of Dual-Threshold Circuits for Low-Voltage Low-Power Applications," *IEEE Transactions On Very Large Scale Integration (VLSI) Systems*, vol. 7, pp. 16–24, 1999.
- [4] Y. Xu, Z. Luo, Z. Chen, and X. Li, "Minimum Leakage Pattern Generation Using Stack Effect," in *ASIC 2003. International Conference on*, 2003.
- [5] M. Keating, D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low Power Methodology Manual*. Springer, 2007.
- [6] R. Vullers, R. Schaijk, H. Visser, J. Penders, and C. Hoof, "Energy Harvesting for Autonomous Wireless Sensor Networks," *IEEE Solid-State Circuits Magazine*, vol. 2, 2010.
- [7] J. Seomun, I. Shin, and Y. Shin, "Synthesis and Implementation of Active Mode Power Gating Circuits," in *Design Automation Conference 2010*, 2010.
- [8] S. Hanson, B. Zhai, K. Bernstein, D. Blaauw, A. Bryant, L. Chang, K. Das, W. Haensch, E. Nowak, and D. Sylvester, "Ultralow-Voltage Minimum-Energy CMOS," *IBM Journal of Research and Development*, vol. 50, 2006.
- [9] K. Shi and D. Howard, "Sleep Transistor Design and Implementation - Simple Concepts Yet Challenges To Be Optimum," in *VLSI Design, Automation and Test. International Symposium on*, 2006.
- [10] R. York, "Benchmarking in Context: Dhrystone," 2002.