

Dynamically Optimizing FPGA Applications by Monitoring Temperature and Workloads *

Phillip H. Jones, Young H. Cho, John W. Lockwood

Applied Research Laboratory

Washington University

St. Louis, MO

phjones@arl.wustl.edu, younghc@gmail.com, lockwood@arl.wustl.edu

Abstract

In the past, Field Programmable Gate Array (FPGA) circuits only contained a limited amount of logic and operated at a low frequency. Few applications running on FPGAs consumed excessive power. Today, the temperature of FPGAs are a major concern due to increased logic density and speed. Large applications with highly pipelined datapaths can ultimately generate more heat than the package can dissipate. For FPGAs that operate in controlled environments, heat sinks and fans can be used to effectively dissipate heat from the device. However, FPGA devices operating under harsher thermal conditions in outdoor environments, or in systems with malfunctioning cooling systems need a thermal management control system.

To address this issue, we had previously devised a reconfigurable temperature monitoring system that gives feedback to the FPGA circuit using the measured junction temperature of the device. Using this feedback, we designed a novel dual frequency switching system that allows the FPGA circuits to maintain the highest level of throughput performance for a given maximum junction temperature. This paper extends the previous work by additionally making this adaptive frequency mechanism workload aware and evaluating power and latency performance under bursty workload conditions. Our working system has been implemented on the Field Programmable Port Extender (FPX) platform developed at Washington University in St. Louis. Experimental results with a scalable image correlation circuit show up to a 30% saving in power for bursty workloads and up to a 2x factor improvement in latency performance as compared to a system without thermal or workload feedback. Our circuit provides power efficient high performance processing of bursty workloads, while ensuring the device always operates within a safe temperature range.

1 Introduction

Many applications are exposed to multiple thermal conditions during their operational lifetime. Mobile systems, such as military and space applications, require high performance computation in embedded systems that move rapidly between different environments. Stationary systems, such as outdoor surveillance systems, must adapt to variable ambient temperatures. Even systems that operate in tightly controlled environments, such as rack-mounted FPGA computational blades in a machine room, must adapt to variable thermal environments so that they will not completely fail due to a fault in a fan or obstruction of air flow. In general, all reconfigurable devices can find themselves exposed to conditions much different than their typical operating conditions. In these cases, it is desirable to allow the circuit to adapt to the environment.

Most existing FPGA circuits operate at a fixed operating frequency. At this frequency, the heat dissipation mechanisms are built to handle worst-case operating conditions. When there is a significant gap between the worst-case operating condition and the typical operating condition, the system must be over-engineered and/or the performance realized by the system may be significantly less than optimal during typical operating conditions.

This work extends upon our previously developed adaptive frequency control mechanism that uses thermal feedback to adjust the operating speed of a reconfigurable system. We have added workload feedback in order to reduce power consumption during application idle periods, and evaluated our approach under bursty workloads against using a fixed frequency with respect to power and latency.

1.1 Motivation

While testing high performance circuits on our reconfigurable development platform, we experienced an incident

*Sponsored by National Science Foundation under grant ITR 0313203.

that overheated an FPGA. Given unfavorable environmental conditions, one of our platforms was damaged because the bitfile generated more heat than the package could dissipate given the amount of airflow available in an open chassis. In order to prevent such an event from occurring in the future, we designed a temperature monitoring circuit that runs on another FPGA within the reconfigurable platform that acts like a thermal circuit breaker. This platform now provides a mechanism to monitor the temperature of the reconfigurable device over the network and provides a mechanism which can dynamically adjust the operation of the reconfigurable logic device.

During our characterization of the FPGA thermal behavior, we discovered that we had an opportunity to make use of the relatively fast measurements of junction temperature changes versus the relatively slow rate of change of temperature of the system due to thermal mass of the package and heatsink. A relatively large amount of time is available to operate a circuit at a high frequency while the package slowly warms as compared to the period at which the platform performs computation on data packets. Seeing this as an opportunity to improve the performance of our reconfigurable hardware platform in transient conditions, we devised a novel scheme that dynamically adjusts the operation of the reconfigurable logic device between two clock frequencies using temperature thresholds. This mechanism generates a thermally-adaptive frequency that maximizes the computational throughput for a specified maximum application temperature, which we refer to in this paper as the application's *thermal budget*. Our current work adds workload feedback to this mechanism and conducts a performance evaluation for bursty workloads.

1.2 Contribution

In the following section, we discuss related academic work and industrial solutions related to thermal management and power management. Section 3 gives a summary of the previous work that we used to build upon in this paper. The main contributions of the previous work was (1) the implementation of a thermal shutdown circuit for applications implemented on FPGAs, (2) a systematic approach for thermal profiling reconfigurable hardware [7], and (3) the development and evaluation of a temperature driven adaptive frequency mechanism to optimize application throughput in response to changing thermal conditions [6]. The contributions of this paper are detailed in Sections 4 and 5. Section 4 extends our previous temperature driven adaptive frequency mechanism to be workload aware, and examines why our mechanism provides power efficient and low latency processing for bursty workloads. Section 5 implements and evaluates the effectiveness of our approach. This evaluation applies our adaptive frequency mechanism to a high power

consumption image correlation application, and quantifies the improvement in power consumption and latency as compared to using a thermally safe fixed frequency for different workload utilizations, burst lengths, and thermal conditions.

2 Related Work

Microprocessors have been built that allow their voltage and frequency to be scaled to extend battery life of mobile computers. Companies that include Intel and AMD have extended this concept to manage heat dissipation on servers [5]. By introducing power management features, software running on the CPU can scale voltage and frequency to lower power usage before the device overheats. Such technology is critical for servers located in large data centers that house hundreds or thousands of computation nodes.

Low-power embedded processors like Xscale [1] have hooks that allow voltage and frequency scaling to manage power. Work presented by [11] makes use of these features to present a dynamic thermal management (DTM) system that scales processor frequency in response to temperature readings from an external thermal couple.

There has also been work in the realm of power management for reconfigurable logic devices. Shang performed power measurement experiments on the Xilinx Virtex-II FPGA to determine the distribution of dynamic power [10]. For the applications analyzed it was found that as much as 22% of dynamic power was consumed by clock resources. Therefore managing the clock tree usage could result in significant power savings. The Virtex-II has entities called BUFGMUXs [12], that can be used for shutting down part of the clock tree or switching to a low frequency during idle times [4]. Meng showed a 25% power savings through low level simulation of a Wireless Channel Estimator application mapped to a Virtex-II, by disabling the clock for portions of the application not in use [9]. One aspect of our paper is to quantify the power savings that can be gained by switching to a low frequency during idle periods for bursty workloads.

3 Using Thermal Feedback

We start this section with an overview of the development platform used for this work. We then summarize our previous work on which this work is built. This consists of a safety thermal shutdown circuit, and a thermally adaptive frequency mechanism.

3.1 Development Platform

The circuits described in this paper were implemented on the Field Programmable Port Extender (FPX) platform,

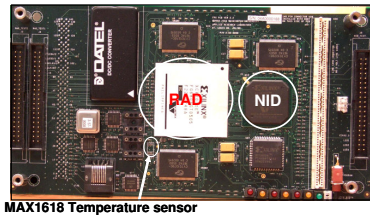


Figure 1. Development Platform

shown in Figure 1. This platform contains two FPGAs: (1) a small Xilinx Virtex FPGA called the *Network Interface Device (NID)* is configured with a static bitfile, and (2) a large Xilinx Virtex FPGA called the *Reconfigurable Application Device (RAD)* is reconfigured with bitfiles loaded dynamically over a network. New bitfiles that implement modular data processing functions are sent to the NID over the network within a bitfile that is used to reconfigure the RAD [8]. The platform uses an on-board Maxim temperature measurement device (MAX1618) to digitally sample the RAD temperature.

3.2 Thermal Shutdown Circuit

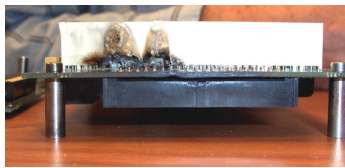


Figure 2. Damaged FPX Platform

Figure 2 shows the side-view of one of our platforms that was damaged by a bitfile running on the RAD that consumed more power than the platform could dissipate in a chassis with insufficient airflow to cool the system. The circuit board warped and caused a short-circuit between power planes.

Motivated by the need to prevent such a high-powered application from damaging another platform, a thermal monitor and shutdown circuit was implemented. The circuit allows the NID to monitor the junction temperature of the RAD. If an application causes the junction temperature of the RAD to surpasses a programmable maximum threshold, then the NID acts as a circuit breaker to unload the high-power bitfile from the device.

Figure 3 illustrates how the temperature monitor and shutdown circuit is mapped onto the FPX. The thermal shutdown circuit was implemented using logic on the NID to prevent an applications deployed on the RAD from exceeding a safe operating temperature. The NID interfaced to a MAX1618, a Maxim temperature monitor chip that

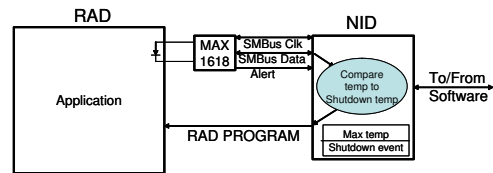


Figure 3. Shutdown Circuit Architecture

measures the junction temperature using a sense diode embedded in the silicon of the RAD. The NID samples the MAX1618 and compares the temperature received from this device to a user-programmable maximum temperature threshold. If the preset threshold is surpassed, the NID shuts down the application deployed on the RAD by sending a command through the SelectMAP interface of the RAD to clear the configuration memory [7].

The temperature of the RAD can also be monitored externally by sending a query message over the network to the NID. The NID responds with a status message that reports the temperature of the RAD. We wrote software to log the temperature of the RAD while running custom-designed thermal benchmark circuits. Section 3.3 discusses how this temperature monitor and shutdown circuit was extended to implement adaptive frequency control of applications deployed on the RAD.

3.3 Temperature Driven Frequency

This section begins with a discussion of the types of applications that benefit from a thermally-adaptive frequency management circuit. Next, we give an overview of our thermally adaptive frequency mechanism. This section concludes with a summary of previous results obtained from applying this mechanism to an image processing application under various thermal conditions.

3.3.1 Target Applications

Reconfigurable systems with certain characteristics benefit most from the use of adaptive frequency control using thermal feedback. First, systems deployed in environments where the temperature changes benefit by allowing the circuit to adapt their performance. Second, systems that have multiple modes of operation that impact their thermal output benefit from adaptive thermal control. Third, systems that have bursty computation with demands for low latency benefit by allowing the device to temporarily operate at frequencies faster than would be allowed in steady-state.

3.3.2 Architecture

Our thermal feedback frequency mechanism is made up of two components; 1) a dual frequency multiplexing circuit,

and 2) a temperature driven frequency controller.

FPGAs available today from vendors such as Xilinx and Altera have Delay Lock Loops (DLLs) that can multiply and divide a clock input signal. We use DLLs combined with a 2:1 multiplexor to implement a dual frequency multiplexing circuit that can switch between the base input clock and a clock that operates at 4x the base frequency. The multiplexer select line determines if the base clock or 4x clock will drive the clock tree. Figure 4 shows the architecture of the Frequency Multiplexing circuit. The 4x clock generation part of this circuit uses the clock multiplier design supplied by the Xilinx XAPP174 [13]. More elaborate techniques can and should be used to avoid clock glitches. For example a glitch free version of the 2:1 mux component can be implemented with the BUFGMUX component available for the Virtex-II [12] and later generations of Xilinx FPGAs.

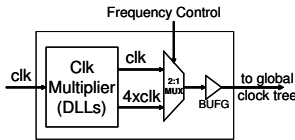


Figure 4. Frequency Multiplexing circuit

The select line of the 2:1 multiplexor is controlled by the temperature driven frequency controller that monitors the application’s temperature and implements a high/low temperature threshold control strategy. Application logic on the reconfigurable device operates using the 4x clock while the temperature remains below the upper threshold. Once the upper threshold is reached, the application circuit is given the base clock and allowed to cool down until the lower threshold is reached. At this point, the cycle repeats.

The main idea of this approach is to modulate the duty cycle at which the application runs with the faster (4x) clock. As the external thermal environment changes, the duty cycle will automatically adjust keeping the application temperature between the upper and lower bounds. By selecting thresholds appropriately and switching quickly between modes, the application can maintain a target average temperature within tight bounds.

The upper temperature threshold is the application thermal budget. The objective is to achieve maximum computational performance for a given thermal budget by adaptively adjusting the duty cycle as the thermal operating environment changes. The mapping of the thermally controlled adaptive frequency mechanism on to our reconfigurable platform is shown in Figure 5. The frequency multiplexer resides in the RAD. The frequency control circuit resides on the NID. This circuit is an extension of the thermal shutdown circuit described in section 3.2. A state machine was developed to implement a temperature threshold

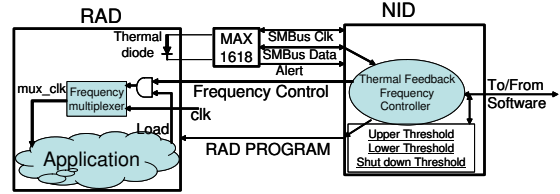


Figure 5. Temperature Controlled Frequency

controller. Configuration commands sent to the NID over the network set the upper and lower temperature threshold values. The thermal budget of the application is the value contained by the upper threshold.

Up to a 2.4x factor improvement in throughput over using a thermally safe fixed frequency was obtained by applying this mechanism to the image processing application described in 5.1. Our previous evaluation used a continuous streaming workload to fully utilize the circuit for several thermal conditions [6].

4 Adaptive Processing of Bursty Workloads

This section first describes the extension made to the thermally adaptive frequency mechanism to make it workload aware. Next the reasons for expecting our approach to be more power efficient and have lower latency for bursty workloads, than using a fixed frequency are discussed.

4.1 Workload Aware Extension

The original temperature driven frequency control mechanism selected between a high and low frequency based solely on the junction temperature of the application FPGA. The underlining assumption being that the application was streaming data from a source that would always fully utilize the available computational resources. There are many cases for which this assumption does not hold, applications with bursty workloads are one such example. When there is no workload to process a natural policy to follow is to run the application at a low frequency. This policy is implemented for our frequency control mechanism by performing an AND of the frequency control signal received from temperature driven frequency controller with a load indication signal generated by the application. Figure 5 shows this AND gate feeding the select of the Frequency Multiplexing circuit.

4.2 Power Efficient Processing

As mentioned in section 2, Shang showed clock resources of circuits evaluated on a Xilinx Virtex-II accounted

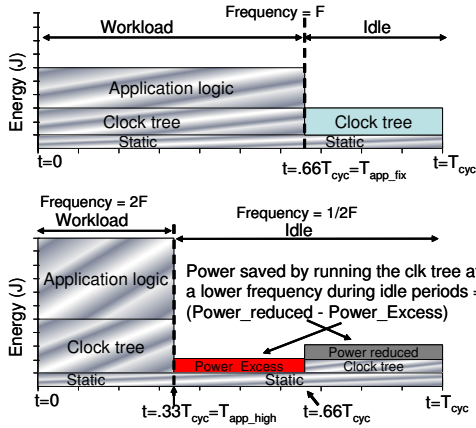


Figure 6. Lower Clock Power During Idle Periods

for 10-20% of dynamic power dissipation [10]. This suggests an opportunity to save power by running an application at a lower frequency during idle periods. The more sparsely loaded an application, the more power saving benefits. Figure 6 shows a graphical comparison between the power usage of an application using a fixed frequency versus a load controlled frequency. In this example the fix frequency is F , and the load controlled frequency switches between a low frequency $= 1/2 F$ and a high frequency $= 2F$. It is assumed that the workload will repeat with a period of T_{cyc} . The workload size for this example is $.66 * T_{cyc}$. The diagonally shaded area represents the power that the fixed frequency and adaptive frequency have in common. Power savings occur for portions of T_{cyc} where the idle time of the adaptive frequency overlap with the idle time of the fixed frequency. Within this region the adaptive frequency is running the clock tree at a lower power than the fixed frequency. The adaptive frequency consumes excess power over using a fixed frequency between the time the adaptive frequency finishes processing a workload burst, and when the fixed frequency would complete processing the burst. Therefore in order to achieve an overall power savings using our adaptive approach, the region of power savings must be greater than the region of excess power usage.

For a given fixed frequency F_{fix} and adaptive frequency F_{adapt} with upper frequency F_{high} and lower frequency F_{low} , a break even workload size (WS_{BE}), can be found such that for workloads sizes less than WS_{BE} using F_{adapt} consumes less power than using F_{fix} . For the configuration used in Figure 6, $WS_{BE} = .66 * T_{cyc}$ ¹. At this point $.33 * T_{cyc}$ time is spent consuming excess clock tree power and $.33 * T_{cyc}$ time is spent saving power. This can be seen by graphical inspection of Figure 6. Analytically the value of WS_{BE} can be found by setting the power used by frequency F_{fixed} equal to the power used by frequency F_{adapt} ,

¹Dynamic power consumption is linearly proportional to frequency

then solving for T_{app_fix} . Equation 1 and 2 are derived from the graphical model shown in Figure 6. They are in terms of quantities that can be directly measured on our reconfigurable platform, and were used to compute power usage in our experimental evaluation (section 5.3).

$$P_{fix} = \frac{P_{load_fix} * T_{app_fix} + P_{idle_fix} * (T_{cyc} - T_{app_fix})}{T_{cyc}} \quad (1)$$

$$P_{adapt} = \frac{P_{load_high} * T_{app_high} + P_{idle_low} * (T_{cyc} - T_{app_high})}{T_{cyc}} \quad (2)$$

4.3 Low Latency Processing

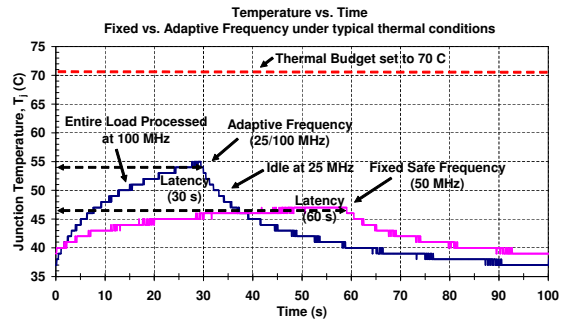


Figure 7. Latency Reduction Example

A load controlled frequency allows an application to make use of excess thermal buffer by running the circuit at a high frequency for a constrained amount of time. If the workload burst length does not cause the circuit to heat to the defined thermal budget, then the burst can be processed completely at the high frequency. Figure 7 illustrates this scenario for a load controlled frequency switching between 25 and 100 MHz, compared to a 50 MHz fixed frequency. If however the burst length causes the temperature to reach the thermal budget of the application, then the temperature controlled aspect of our approach, section 3.3, sets up a duty cycle between the high and low frequency to process the rest of the burst at an effective frequency that is near optimal for the current thermal conditions. Figure 14 of section 5.3 gives an illustration and discussion of this scenario.

5 Implementation

This section first describes a computationally intensive circuit implemented on an FPGA that is capable of exceeding the safe thermal limits of the FPGA package of 85 C. Next we apply and evaluate our adaptive methods using this application in a case study.

5.1 Image Correlation Application

Image correlation is an application well-suited for hardware implementation. It is highly parallelizable [3, 2]. The specific image correlation application we use in our performance evaluation scans an input image for up to four different patterns. The circuit is inherently high-powered and cannot run at its maximum clock rate without thermal management or it overheats the FPGA.

The core logic of this application was used to evaluate the effectiveness of using thermal and load frequency control. Instead of reading image data from external memory, signals from a block RAM and a Linear Feedback Shift Register (LFSR) were used to produce pseudo-random data for the core to process. Results of synthesis and characteristics of the application are given in Figure 8. Further implementation details of this application can be found in [6]

VirtexE 2000 Resource Utilization				
Lookup Tables (LUTs)	D Flip Flops (DFFs)	Occupied Slices	Block RAM	Max Frequency
72% (27,788)	64% (24,832)	82% (15,808)	26% (43)	125 MHz

a.)

Image Correlation Characteristics				
Image Size (# pixels)	Pixel Resolution	# of Mask Patterns	# of Templates	Image Processing Rate
640x480	8-bit (grey scale)	1 - 4	10 (in parallel)	12.7/second (at 125 MHz)

b.)

Figure 8. a.) FPGA Usage, b.) Application Details

5.2 Experimental Setup

The image correlation application is deployed on the RAD FPGA of the FPX platform. This platform was installed into a 3U rackmount case. The case is equipped with 2 fans that each supply approximately 250 Linear Feet per Minute (LFM) of air flow.

Evaluation experiments were performed between using our temperature and load control frequency approach and using a fixed frequency. These experiments were conducted under two different thermal conditions for a set of different workload sizes and burst lengths. Figure 9 describes the two thermal conditions and Figure 10 gives details of the workload characteristics. The fixed frequency used in these experiments was determined by finding the frequency, under worst case thermal conditions, at which a thermal budget of 70 C for a continuous workload would be maintained. This frequency was found to be 50 MHz and is referred to as the thermally safe fixed frequency for the application. The adaptive frequency was configured to switch between a low frequency of 25 MHz and a high frequency of 100 MHz.

Thermal Condition	Ambient Temperature (C)	# of Fans
Typical	25	1
Worst Case	35	0

Figure 9. Evaluation Thermal Conditions

Work Load Size (% of 100 second Cycle Period processing images)	Burst Length (# of consecutive Images)			
	1	10	100	300
.2	1	N/A	N/A	N/A
2	1	10	N/A	N/A
20	1	10	100	N/A
60	1	10	100	300

Figure 10. Work Load Characteristics: Workload Size is % of Cycle Period spent processing images, using a 50 MHz frequency

5.3 Results and Analysis

Figures 11 and 13 provide a summary of the performance evaluation results. It was found that up to a 30% reduction in power and a up to a 2x factor improvement in latency was achieved using our adaptive frequency approach compared to using a thermally safe fixed frequency. The following gives a discussion of these results; first in terms of power usage, and then from a latency perspective. This section concludes with an examination of how burst length impacts the thermal behavior of the image correlation circuit.

5.3.1 Power

Frequency (MHz)	Average Power (W)					
	WS=.2%	WS=2%	WS=20%	WS=60%	WS=80%	WS=100%
Fixed (50)	1.76	1.81	2.34	3.51	4.10	4.68
Adaptive (25/100)	1.20	1.27	1.92	3.38	4.10	4.83
Power Savings	31.9%	30.2%	19.9%	3.8%	0%	-3.2%

Figure 11. Average power comparison

Figure 11 shows the power usage measured for the fixed and adaptive frequency for different workload sizes. Workload size is defined to be the percent of time needed by the 50 MHz fixed frequency to process the images it receives for a 100 second Workload Cycle Period. Experiments were run for workload sizes from .2% to 60%. Power numbers for workload sizes of 80% and 100% were extrapolated. Burst length is not considered because it does not impact power consumption as long as the thermal budget of the circuit is not reached. If the thermal budget is reached, then the adaptive frequency will operate at a lower effective frequency, which in turn will cause power consumption to drop. Therefore the numbers given in Figure 11 are an upper bound for the power consumed by the adaptive load controlled approach. This approach uses 31.9% less power

than the fixed frequency for the smallest workload size of .2% and saves 3.8% for the largest workload size considered. Extrapolating for larger workload sizes shows that our approach will give power saving for workload sizes less than 80%², and will at most use 3.2% more power than a 50 MHz fixed frequency for a workload size of 100% (continuous workload). Given workload sizes greater than 50% are beginning to look more like continuous workloads than bursty workloads, our results show that this approach is well suited for workloads that are highly bursty.

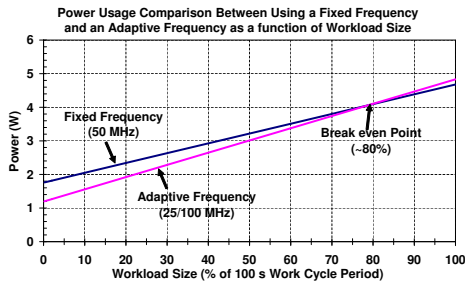


Figure 12. Power versus Workload Size

5.3.2 Latency

Latency Comparison
(Thermal Condition = 1 Fan, Ambient Temperature 25 C)

Frequency (MHz)	Latency (s) (Burst Length = Workload Size)			
	WS=.2%	WS=2%	WS=20%	WS=60%
Fixed (50)	.2	2	20	60
Adaptive (25/100)	.1	1	10	30
Improvement Factor	2	2	2	2

a.)

Latency Comparison
(Thermal Condition = no Fan, Ambient Temperature 35 C)

Frequency (MHz)	Latency (s) (Burst Length = Workload Size)			
	WS=.2%	WS=2%	WS=20%	WS=60%
Fixed (50)	.2	2	20	60
Adaptive (25/100)	.1	1	10	34.2
Improvement Factor	2	2	2	1.75

b.)

Figure 13. a. Typical, b. Worst Thermal Condition

Figure 13 gives a summary of the latency measurements obtained for using an adaptive versus fixed frequency for two thermal conditions. For all but one experimental setup the adaptive approach shows a 2x improvement in latency performance over using a fixed frequency. The Worst case thermal condition shows a 1.75x improvement for the largest workload size considered. Reaching the 70 C thermal budget before the workload burst completes processing causes the reduced performance. This is shown clearly in

²Break even point is 80% instead of expected 66.6% because the measured power consumption for the workload processing at 100 MHz was 2% less than linear extrapolation predicts

Figure 14. The bottom plots of this figure show the thermal behavior of the fixed and adaptive frequency for Typical thermal conditions and a Workload Size = Burst Length = 60%. As expected the peak temperature reached by the adaptive frequency is higher than the fixed frequency. Under Typical thermal conditions even for a fairly large burst size there is a significant thermal buffer between the adaptive frequency peak temperature and the 70 C thermal budget. The top plots show the same workload scenario under Worst case thermal conditions. For this case the adaptive frequency reaches the thermal budget before the workload completes processing. Upon reaching the thermal budget the thermally adaptive component of our approach, section 3.3, begins to switch between 25 and 100 MHz to cap the junction temperature at 70 C until processing of the workload completes. This results in the latency increasing from 30 seconds to 34.2 seconds, a 14% increase, which is still a 1.75x improvement in latency over using the thermally safe fixed frequency.

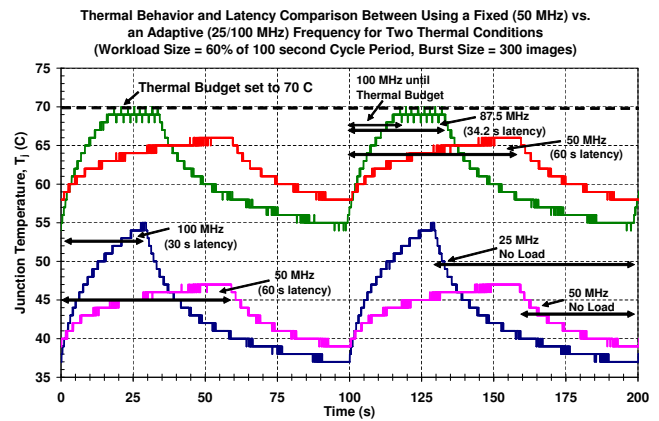


Figure 14. Thermal and Latency Comparison

5.3.3 Burst Length Impact on Thermal Behavior

Frequency (MHz)	Temperature (C) (Min/Max) Thermal Condition: 1 Fan				Temperature (C) (Min/Max) Thermal Condition: no Fan			
	BS=1	BS=10	BS=100	BS=300	BS=1	BS=10	BS=100	BS=300
Fixed (50)	42/43	42/43	42/46	39/47	61/62	61/62	59/64	58/66
Adaptive (25/100)	41/42	41/43	40/50	37/55	60/61	60/62	56/67	54/70

Figure 15. Burst Length Impact on Thermals

In addition to conducting experiments with different workload sizes, each workload was broken into several different burst lengths. For example for workload size = 60% the workload may be processed as burst lengths of 1 image, 10 images, 100 images or 300 images (burst length =

workload size). Figure 15 shows how the steady state maximum and minimum temperature changes as the burst length is varied. Figure 16 shows this information as a plot for the Worst case thermal condition. The burst length used to process a given workload size has a large impact on the thermal behavior of the application. As an example, under Worst case thermal conditions a burst length of 300 images causes the application to heat up to the 70 C thermal budget, thereby causing an increase in processing latency. If the workload was broken into evenly spaced bursts of 1 image, then the maximum temperature would only reach 61 C. The same amount of work is done for the Workload Cycle Period, however, spreading the processing across the entire Workload Cycle Period as small bursts allows each image to process with minimum and constant latency. This knowledge of thermal behavior would be important for applications where constant latency is important, such as streaming media applications.

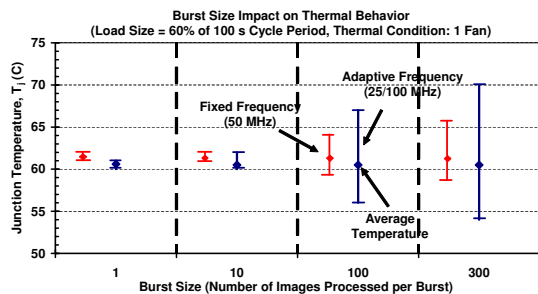


Figure 16. Burst Length Impact on Thermals

6 Conclusion

A low latency and power efficient approach was presented for processing bursty workloads in reconfigurable hardware. Our adaptive approach safely manages the use of excess temperature margins to increase processing speed while an application is under a workload, and conserves power by reducing an application's clock rate during idle periods. Performance evaluation experiments with a scalable image correlation circuit show up to a 30% savings in power for bursty workloads and up to a 2x factor improvement in latency performance as compared to a system without thermal or workload feedback.

References

[1] *Intel 80200 Processor based on Intel XScale Microarchitecture Developer's Manual*, 2003.
 [2] K. Chia, H. J. Kim, S. Lansing, W. H. Mangione-Smith, and J. Villasenor. High-performance automatic target recognition through data-specific vlsi. *IEEE Transactions on Very Large Scale Integration Systems*, 6(3):364–371, Sept. 1998.

[3] Y. H. Cho. Optimized automatic target recognition algorithm on scalable myrinet-field programmable array nodes. In *34th IEEE Asilomar Conference on Signals, Systems, and Computers*, Monterey, CA, Oct. 2000.
 [4] S. Choi, R. Scrofano, V. K. Prasanna, and J.-W. Jang. Energy-efficient signal processing using fpgas. In *FPGA '03: Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays*, pages 225–234, New York, NY, USA, 2003. ACM Press.
 [5] Intel Corporation. Addressing power and thermal challenges in the datacenter, 2005.
 [6] P. H. Jones, Y. H. Cho, and J. W. Lockwood. An adaptive frequency control method using thermal feedback for reconfigurable hardware applications. In *IEEE International Conference on Field Programmable Technology (FPT)*, Bangkok, Thailand, Dec. 2006.
 [7] P. H. Jones, J. W. Lockwood, and Y. H. Cho. A thermal management and profiling method for reconfigurable hardware applications. In *16th International Conference on Field Programmable Logic and Applications (FPL)*, Madrid, Spain, Aug. 2006.
 [8] J. W. Lockwood, N. Naufel, J. S. Turner, and D. E. Taylor. Reprogrammable Network Packet Processing on the Field Programmable Port Extender (FPX). In *ACM International Symposium on Field Programmable Gate Arrays (FPGA '2001)*, pages 87–93, Monterey, CA, USA, Feb. 2001.
 [9] Y. Meng, W. Gong, R. Kastner, and T. Sherwood. Algorithm/architecture co-exploration for designing energy efficient wireless channel estimator. *Journal of Low Power Electronics*, 1:238–248, 2005.
 [10] L. Shang, A. S. Kaviani, and K. Bathala. Dynamic power consumption in virtex-ii fpga family. In *FPGA '02: Proceedings of the 2002 ACM/SIGDA tenth international symposium on Field-programmable gate arrays*, pages 157–164, New York, NY, USA, 2002. ACM Press.
 [11] E. Wirth. Thermal management in embedded systems. Master's thesis, University of Virginia, 2004.
 [12] Xilinx. *Virtex-II Platform FPGA User Guide*, 2005.
 [13] Xilinx Inc. Using delay-locked loops in spartan-ii fpgas. Xilinx XAPP174, Jan. 2000.