# Candidate Statement

**Aviral Shrivastava**, School of Computing, Informatics, and Decision Systems Engineering

**Overview**

*My research interests lie at the interface of processor design and software to build better embedded computing systems.* Since the beginning of my appointment, I have pursued a set of integrated activities for research, teaching, and service in the area of **architectures and compilers of embedded systems**.

The significance of my research can be estimated through the output of my work. I have co-authored 1 book, 2 book chapters, 19 journal publications (17 in IEEE and ACM Transactions), 51 highly selective, high-impact, archived and refereed conference publications (in top embedded systems conferences including CODES+ISSS, CASES, and LCTES)[1]. My work has been cited more than 393 times, with an h-index of 10[2]. My book has been adopted as recommended reading[3] in undergraduate courses, and my papers are part of reading material in graduate courses[4] in top universities. I have been invited by several universities and companies to present my research. In particular, my lecture at Microsoft Research is here. I am the recipient of the prestigious **2010 NSF CAREER Award**, and in the last five years I have obtained **personal external funding of about $1.5 million** (investigator recognition) from federal and state funding agencies, including NSF and SFAZ, and industry, including Microsoft Research and Raytheon Missile Systems. The scale of these projects and the significance of my role in them can be inferred from the fact that the **total award amount received is about $3.4 million**. This funding has enabled me to establish Compiler Microarchitecture Lab (CML) at ASU, with a mission to "*develop technology and tools at the compiler-microarchitecture interface to solve the challenges faced by embedded system designers*". In this lab, I have mentored 1 postdoctoral associate (now Assistant Professor at UNIST, South Korea), 7 M.S. students (with excellent job placements including Intel, Microsoft, and nVIDIA). Currently I am mentoring 5 doctors, 8 masters, and 1 undergraduate student. My first Ph.D. student is slated for graduation in fall 2011.

Education is an essential component of academic research. As an Obama mentor I am firmly committed to excellence in teaching at both the undergraduate and graduate levels. I introduced a new graduate level course "CSE 591: Low Power Computing" to identify and train students in my area of research. A very high percentage of graduate students who take this course continue to finish their Master's thesis with me. I am also actively involved in undergraduate education in our department through my teaching and curriculum development activities. Teaching at regular load, I have taught 3 different core undergraduate courses. I have revamped 2 of the 3 courses to reflect the recent shifts in computer architecture and programming. I refocused CSE 230 to emphasize on computer design rather than assembly programming, and introduced modern multi-core architectures in CSE 420. The effectiveness of my teaching pedagogy is reflected in the encouraging reviews from students. The average teaching evaluation for my five years at ASU is 4.30/ 5.00.

I believe that the success of a university like ASU requires active support and involvement of its faculty. I have been involved in the school level committees, including graduate admissions committee, computing resource committee, and undergraduate program committee. I have also actively provided technical and professional service outside ASU, not only as program committee member of major embedded system conferences, but also in their organization. I am the web chair of ESWEEK (the most important embedded systems event) for the past two years. Last year, we hosted ESWEEK 2010 in Phoenix. In addition I am the general chair of the "Compiler Assisted Soc Assembly" CASA 2011 workshop.

---

[1] Conferences are much more prestigious and have higher impact than journals in Computer System Engineering.
[2] An h-index of 10 implies that at least 10 of my papers have been cited at least 10 times each.
[3] My book, "Power-Efficient System Design," has been adopted as a reference book for an undergraduate course in UCLA.
[4] My paper, "An Efficient Compiler Technique for ...," was required reading in a graduate course at UCLA

My last five years at ASU have been energizing and productive. I believe I have substantially raised ASU profile both externally and internally. The following sections provide a summary of my main accomplishments in research, teaching, service, and the synergies between them.

**My Research Questions**

Computing has permeated our lives inside-out, e.g., implantable, cardiovascular defibrillators, or "emergency room in the chest," inside our bodies, to cell phones, TVs, washing machines, etc., very close to us, and data warehouses, cloud computing etc., which we access remotely. This unprecedented integration of computing in our lives has been made possible because of two main reasons. **i) we are able to design very power-efficient computers, and ii) very reliable computers.**

The earliest of computers, e.g., UNIVAC used 66 Watts of power to perform one arithmetic operation. We have come a long way from there. Today, a typical desktop processor can perform 58 million operations per second in a Watt of power. However, there is need to further improve the power-efficiency of computation to solve some of the most important scientific and engineering questions that we face today. For example, modeling climate, exploring high power-density bio-fuels, simulating fusion reaction etc., require exa-scale computing ($10^{18}$ operations per second). Exascale is primarily limited by power-efficiency of computation. Consider this: today the state-of-the-art processor can perform $10^9$ operations per second in a Watt of power. To achieve exa-scale computation, we would need 1 billion such components, but, they would consume 1 billion Watts of power. Clearly this is not feasible. Enabling exa-scale computing requires improving power-efficiency of computation by two orders of magnitude. Higher power-efficiency is needed not only for high performance computing, but also for embedded computing, where battery determines the weight, size, shape, recharge time, frequency, or essentially the usability of embedded system. Thus one focus of my research is on "**how to make computers even more power-efficient?**"

Reliability is another glorious achievement of computing. Consider the fact that today's computers are composed of about a billion transistors, each switching a billion times in a second. In other words, one quintillion switches per second happen reliably to produce the correct output for years of a computer life. It is because of such high levels of reliability that computers have found extensive use in safety and life-critical applications, flight controllers, and pacemakers. However, this golden era of reliability is coming to an end. As we shrink the transistors (to improve performance and power-efficiency), they become increasingly susceptible to soft errors. Soft errors are mainly caused by cosmic radiation particles falling on the chip. As transistors become small, even a low-energy particle strike can toggle the value of a transistor, causing soft error. Juxtapose this with the fact that there are exponentially more particles with lower energy than those with higher energy in the atmosphere. Consequently, even though soft error rates now are about once-per-year, experts predict that in a decade or two soft error rate will be about once-per-day. Therefore, the second focus of my research is: "**reliable computing on unreliable hardware**".

**Current and Continuing Research**

I am excited to be a part of one of the most important revolutions in computer systems design: the transition from single-core to multi-cores. Multi-cores provide the only way to continue increasing performance without much increase in the power consumption. However, the real challenge in multi-cores is the difficulty of programming them. In fact, famous computer science Professor, and current president of Stanford University John Hennessy noted in his keynote at ESWEEK 2010 that programming multi-cores is "*one of the biggest challenges that computer science faces today.*"

*A significant portion of my efforts have been, and will continue to be towards* ***developing compilers for multi-core architectures, simplifying their programming, and therefore enabling their use.*** I am working on developing compiler tool and techniques for several promising multi-core architectures:

- Purely Distributed-Memory Multi-core Architectures: This very promising architecture is used in the IBM Cell processor, which is used in Sony Playstation3, and is also used to build Roadrunner, the first petaflop ($10^{15}$ operations per second) machine. We are developing compiler tools and technology for the Cell processor. This project has been supported by grants from NSF and SFAZ, and has brought us a lot of visibility, both in the academic and industrial arena. Our papers on this have been accepted in top embedded systems conferences and journals, and I have been invited by companies, including LSI and TI, and "by invitation only" workshops to talk about this. In addition, we are helping out Rainbow Studios, a local game development company in parallelizing games on the Cell processor.
- Graphics Processing Unit (GPU) Computing: GPUs have truly brought supercomputing to the masses. GPUs typically have hundreds of cores, and are present in almost all desktops and laptops. The biggest challenge in compiling for GPUs is using the memory hierarchy efficiently. Our recent paper in DAC 2011 presents an automatic technique to analyze the effect of memory hierarchy on performance of GPU architectures. We received a donation of 2 Tesla C2060 and 3 Fermi cards from Nvidia for this research.
- Non-Coherent Cache Architecture: Recent 48-core Single Chip Cloud computer (SCC) from Intel is a non-coherent cache architecture. Our proposal to develop compiler techniques for the 48-core was accepted, and now we one of handful of universities in world that have access to this chip for research.
- Coarse Grain Reconfigurable Arrays: While multi-core architectures improve power-efficiency, they can only take it so far. Even higher power-efficiencies can only be achieved through acceleration. Towards that, we are developing programmable accelerators, called Coarse Grain Reconfigurable Arrays or CGRAs and their compilers. We are among the leaders in CGRA compiler research with the best application mapping technique, and are credited for being pioneers in memory-aware compilation for CGRAs. Our application mapping technique paper was a best paper candidate in ASPDAC 2008. I was invited to a special panel on CGRA compilation in CODES+ISSS 2010.

Since soft errors are a hardware phenomenon, most research on protection from soft errors till now has been at the hardware layer. But hardware techniques are running out of steam. This implies that soft errors will be exposed to software. Instead of exposing them to the programmer, we want to contain them in the compiler, thereby reducing programmer's burden. We are the pioneers in using compiler techniques to protect programs from soft errors. Our papers on this topic have been accepted at LCTES 2009, DATE 2010. Out paper on a static analytical technique to estimate vulnerability of processors to soft errors was the **second highest ranked paper at LCTES 2010**. I was also awarded the **2010 NSF CAREER award for teaching and research in "Compilers for Soft Errors"**.

**Teaching**

For me, teaching and interacting with students is one of the biggest attractions of an academic career. The experience gained from instructing students has been extremely rewarding and invaluable. Indeed, through my past interactions with students, I have gained insights into new ways of approaching problems and learned how to present information in a more understandable way. My undergraduate and graduate level class evaluations have been consistently higher than average with some excellent reviews from students, and an overall instructor rating of 4.30 out of 5.

I developed a new graduate level course: CSE591: Low Power Computing" in spring 2007. I have been teaching this course every spring semester. This is a research course in which we discuss some of the most important papers published in the last two years, and students do a research project with a goal of producing a publishable quality paper. Indeed this course is a lot of work for me, as I every time I have to

read new papers and have to brainstorm with the students about what novel projects they can do, that can turn into publishable quality work. However, the rewards are worth it. We submit some of the top research projects to international conferences, and every year one or two of them get accepted! Students have acclaimed this to be an "amazing course", and say that they "need more classes like this in the graduate program." Students also say that the "goal of producing a publishable-quality paper is extremely good experience, and lends itself to preparation for choosing and researching a topic for a graduate thesis or project." In spring 2009, one of the students in the class did a project on "Power-Accuracy Tradeoffs in Human Activity Transition Detection," in which he developed a low-power scheme to monitor the activity of stroke patients at home. This work was accepted at the prestigious DATE 2010 conference.

In addition to graduate education, I have made significant contributions to undergraduate education in our department through my teaching, course and curriculum development activities. Undergraduate classes are harder to modify, because structure and organization is the most important thing for undergraduates. Every change must be thoroughly integrated into the big picture. However, I think it is extremely important to continue to update the undergraduate curriculum, in the wake of fast developing computer science. When I started, the focus of the undergraduate class "CSE 230: Computer Organization and Assembly language" was on assembly language programming. Today there is little value in learning assembly language programming, because firstly it is just another programming language, and secondly, thanks to advanced compiler technology programming in assembly language is now minimal. I reorganized this class to delve more into the organization of the processor. Instead of just learning about processor, students design processor simulator in my undergraduate class, and that gives them the "aha" feeling. They really get to understand, how computers are built.

In order to prepare the students to compete in the multi-core computing world, I revamped CSE 420 to increase the emphasis on the multi-core architectures, instead of scheduling policies in the 30-year old, IBM 360. In this class, I teach students how to program multi-cores and give programming assignments on Sony Playstation 3 houses the IBM cell processor which has 9-cores. Students play games on PS3, but this is their opportunity to program it, and students are excited about it. Students were very excited about programming on a cutting-edge platform, and voted it as the best assignment in the class, and commented, "Very, very interesting material and has such far-reaching implications (towards the end of the material) that this class is almost too much," and that "It was interesting to see how processors work and the current challenges facing the processor industry."

**Service**

I have made continuing, productive contributions to ASU. Over the years, I have served on the i) Graduate Admissions Committee, ii) Undergraduate Program Committee, and iii) Computing Resource Committee.
I have had an active technical and professional service outside ASU: I have served in the program committee of all major embedded systems conferences, including CODES+ISSS, CASES, and LCTES. I regularly chair sessions in these conferences. In addition, I have been the web chair of ESWEEK (CODES+ISSS, CASES, and EMSOFT) in 2010 and 2011, and have organized the Works-in-Progress Session at LCTES in 2009 and 2010. For 2011, I have been invited to organize the prestigious CASA 2011 workshop. My future goal is to organize recognized international conferences in my research areas, and become an editorial board member of prestigious journals.

My last five years in the School of Computing, Informatics and Decision Systems Engineering have been exciting, energizing and productive. In the coming years, I strongly believe that my current and future teaching and research efforts will increase the significance and visibility of the School of Computing, Informatics and Decision Systems Engineering, Fulton Schools of Engineering and Arizona State University. **I look forward to continuing my career here towards an internationally renowned research program.**