DPA26BZ02-DV010
Low Resource Computing
Frequently Asked Questions (FAQs)

1. Is there a preferred operational domain or mission context, or is the program intentionally domain-agnostic?
A. The program is intentionally domain-agnostic. DARPA is interested in proving that this approach isn't just tied to one platform (e.g, UAS), but can address a wide range of DoW needs and legacy platforms.

2. The solicitation mentions "semantic overlays" and microprogramming: are these intended as illustrative examples of the kind of approach you're looking for, or are you open to other injection techniques?
A: We are open to, and encourage, other techniques both for injection of the capability and for the computational optimization itself.

3. For device selection, is there an era or decade of legacy hardware introduction that you have in mind for the demonstration? This would help us narrow down the relevant range for low-memory requirements.
A: As you may know, many of the Department of War's (DoW) foundational computational, sensor, and navigation systems are currently operating on hardware ranging from 10 to over 40 years old.

4. For the Phase I feasibility demonstration, is a publicly available dataset acceptable for benchmarking, or do you expect proposers to obtain access to data on actual fielded hardware?
A: We have to defer to the language in the SBIR call here, but there is no specific requirement stated for Phase 1 of basing the comparison off of data you collect from real-world devices in the field.

5. For Phase II's three-platform requirement: do the three platforms need to represent meaningfully different system classes, or is demonstrating the same capability on three devices within the same architectural family acceptable?
A: We have to defer to the language in the SBIR call here, and there is no requirement for them to be across meaningfully different system classes. Of course, the SBIR aims to show results on multiple systems to show that this approach is usable - and so more variation helps communicate that more clearly to DoW stakeholders.

6. What specific resource threshold must a target platform meet to qualify as "low resource"? Is the <25% criterion satisfied by any single dimension (RAM, CPU, or disk) or must it apply across all three?
A: To qualify as "low-resource," the target platform must be end-of-life and/or possess less than 25% of the proposed new-system resources. The solicitation states this <25% criterion is satisfied if it is met in at least one dimension ("in one-of RAM, CPU, disk").

7. In addition to resource constraints, is there importance for "usefulness" of any upgraded capability (e.g., 2x better track accuracy, etc.)? And should "usefulness" metrics like these also be verified at a certain TRL level in the Phase I and follow-on Phase II?

A: While the solicitation does not mandate specific numerical thresholds for "usefulness" (e.g., 2x better accuracy) or specific Technology Readiness Levels (TRLs), it requires demonstrating "mission-capable performance" and bringing "cutting-edge capabilities" to a system where it was previously believed impossible. Crucially, the usefulness and feasibility case must prove that delivering the capability on the hardware was either (1) not feasible or (2) would take over 2 years of effort. The realism of this case will be evaluated by the Government.

8. For FPGA- or DSP-based platforms where RAM/CPU/disk metrics don't map cleanly, what equivalent resource metrics (logic elements, DSP slices, on-chip SRAM, throughput) will be acceptable for demonstrating the resource constraint qualification?
A: The document acknowledges non-standard architectures by allowing the <25% threshold to be met via "the equivalent in other device architectures e.g. FPGAs, DSPs." It does not specifically list acceptable equivalent metrics (like logic elements or DSP slices), leaving it to the proposer to define and justify the equivalent resource dimension in their Phase I analysis. I think utilization of a number of DSP slices, taking into account hard logic elements, is a good one. Creativity in using otherwise-unused hard logic elements is certainly a possibility.

9. What technical techniques are acceptable for achieving "semantic overlays" or meta-programming on legacy hardware? Specifically, are hardware-aware neural network compression/quantization, binary translation, firmware-level virtualization, or RTOS optimization considered eligible approaches alongside the meta-programming method described?
A: The solicitation highlights "meta-programming borrowed from the security community" and "semantic overlays" as the primary examples. It does not explicitly mention or exclude hardware-aware neural network compression, binary translation, or RTOS optimization. However, it welcomes "innovative solutions that repurpose existing hardware to add net-new features" and emphasizes creatively applying computational resources. The only explicitly forbidden approach is creating new chips or new computing architectures. Firmware-level virtualization is, as we understand it from your question, likely to be a type of meta-programming, as could be some types of RTOS optimization. However, those are just examples we included – we look forward to seeing innovative suggestions.

10. What security compliance posture is required for the re-purposed low-resource system? Should it meet a specific framework (e.g., NIST SP 800-171, RMF ATO) during Phase I, or is security compliance only required to be demonstrated by Phase II?
A: The document requires the effort to "demonstrate achieving mission-capable performance, security, and stability." It does not stipulate adherence to any specific compliance frameworks (such as NIST SP 800-171 or RMF ATO) for either Phase I or Phase II. However, some security compliance in Phase II would support it's goal of being shown as able to be transitioned into a real system.

11. For the Phase I prototype demonstration, what level of operational realism is required? Laboratory bench test, hardware-in-the-loop (HWIL), or field environment? And should it be witnessed/validated by a Government representative?

A: Phase I requires a "demonstration of prototype system with initial performance metrics documented" by Month 5. It does not mandate a specific level of operational realism (e.g., HWIL or field testing) or explicit witnessing by a Government representative for Phase I. By contrast, Phase II (Option) specifically introduces "integration with program-provided hardware-in-the-loop (HWIL) test-bed as applicable" by Month 6. Proposers are also encouraged to use actual existing low-resource devices for their prototypes, though a surrogate may be proposed if needed. We do plan to witness/validate these solutions, as soon as performers are ready, in either phase.

12. To better align our proposal with the program's end goals, are there any additional requirements, priorities, or recommendations that are not fully addressed in the solicitation?
A: While all formal requirements are contained within the solicitation, a key recommendation to ensure alignment with the program's objectives is that any target device chosen for the demonstration should not be produced by the proposing vendor. The fundamental intent of the Low Resource Computing program is to demonstrate the ability to extend, optimize, and secure a fielded legacy device where the performing team lacks inherent OEM control, proprietary design data, or source code access. Selecting a third-party, "black-box" legacy system as your demonstration target is critical to proving that your proposed LRC techniques can be broadly applied to fielded DoW assets without relying on proprietary vendor insights.

13. What kind of demonstrations, examples, or situations are preferred to be addressed in the proposal?
A: Because the primary objective of the LRC program is to enable the reuse of DoW assets by hosting modern capabilities on legacy or resource-constrained hardware, proposals must address LRC as a system-level challenge. Isolated component optimizations are compelling only if they unlock broader system-level benefits, enable massive hardware reductions, or operate at extreme performance margins.
For example, consider an optimization for a signal processing module within a legacy airborne tactical pod that draws 500W:

- **Inconsequential: Simply reducing that specific module's power consumption from 5W to 100mW is largely inconsequential to the pod's overall 500W power profile.**
- **Compelling (Capacity Expansion): That optimization becomes highly compelling if the resource reduction frees up enough CPU/RAM margin on the legacy hardware to run new, high-value mission software (e.g., modern AI-driven target classification or advanced sensor fusion) without requiring hardware upgrades.**

Beyond capacity expansion, other compelling, system-level demonstrations of operating under extreme constraints include:

- **Subsystem Elimination & Extreme Longevity: A system operating at such a low power threshold that it can completely eliminate the components (such as degrading chemical batteries) that would otherwise preclude it from operating independently for a hundred years.**
- **Extreme Reliability Tolerance: A system designed to tolerate such inherently low hardware reliability that useful computational progress can still be made even with a mean-time-between-reboot (MTBR) of 10 milliseconds.**

Note: While these examples illustrate how to think through system-level impacts across compute, power, and reliability, they are not restrictive. DARPA highly encourages creative proposals that bring modern capabilities to legacy systems across any critically constrained resource.