By Joseph C. Libby and Kenneth B. Kent

An Embedded Implementation of the Microsoft Common Language Infrastructure

The Common Language Infrastructure (CLI) provides a framework for managing and executing applications. [1] Developers designing applications for the CLI need not worry about the underlying architecture as it is abstracted from view by the CLI framework. This abstraction, while a boon for developers, leads to degraded performance. It is because of these inefficiencies that the CLI is not well suited for developing embedded applications. It would, however, be beneficial for developers to be able to develop embedded applications using the CLI. In order to address the issues caused by the extra layer of abstraction added by the CLI, an embedded processor is designed and implemented. This processor is capable of natively executing the CLI instruction set which effectively removes the performance problems caused by the extra layer of abstraction.

Benchmarking results show that the embedded CLI processor implemented for this project is capable of outperforming the software CLI on the benchmark applications used. These results are promising as the goal of this project was to increase the performance of the CLI through native execution.

1. ECMA (European Computer Manufacturers Association) International, Standard ECMA-335 (4th Edition): **Common Language Infrastructure (CLI) Partitions I to V**, June 2006 .