
Abstract

The Message Passing Interface (MPI) standard makes it possible to write portable software for multi-processor computers. Programs complying with it can be ported between massively parallel processors and networks of workstations without consideration for the underlying hardware. An embedded system that supports MPI allows programmers to develop, debug, validate and characterize parallel algorithms on workstations. Only performance tuning need take place on the target hardware. Development can proceed before the hardware even exists and more programmers can share what might remain a scarce resource once it does exist.

This paper describes Lincoln Laboratory embedded MPI (LL/eMPI), an implementation of a subset of standard MPI designed to be small enough to use in embedded systems with limited memory. We discuss the subset and our reasons for choosing it and then describe our target hardware and our implementation's structure. Finally we present results that show that the implementation is small and fast.