Front Matter

# Acknowledgements

I would like to thank my dissertation advisor Dr. Nutter for his support of my project. His continual guidance has helped keep my project on track, and has given me countless inspirations for my project. I would also like to thank Dr. Andersen and Dr. Cooke for introducing me to so many interesting topics in Computer Science that were incorporated in my project. My thanks also extend to Dr. Mitra and Dr. Pal for their support of my project. Elliot Briggs, a fellow graduate student in Electrical Engineering, also has my thanks for his help in assembling the hardware in the project.

I would like to thank the Electrical Engineering department and Dr. Mitra for their financial support over the years, without which I would not have been able to achieve all that I have achieved.

I am very grateful for the support of my friends and family over the years. They have always shown support for my endeavors and encouraged me to reach my highest potential. To my loving wife, Melissa; I am so lucky to have such a wonderful wife who put up with my long hours, constant prattling about my project, and always supported me.

# Abstract

Parallel computing is currently undergoing a transition from a niche use to widespread acceptance because of the recent availability of multi-core processors in the personal computing market. These processors have given new levels of performance to desktop applications, allowing algorithms that were once the domain of supercomputers to now be run on home systems. The embedded world has yet to embrace these parallel systems except in ultra-high-end systems. Network performance is critical in parallel computing, and the routing algorithm has a direct impact on performance. In this dissertation, a prototype toolkit is developed consisting of five primary parts: a prototype router implemented using TI F2808 DSP controllers, a physical and data link layer that utilizes SPI as a base, a network layer protocol stack for handling transmission across a network, a routing algorithm, and an API that simplifies the programming of parallel algorithms. The routing algorithm is based on the seminal work of Harald Räcke. Räcke's method is very good at avoiding congestion, but is generally too complex for embedded systems. The algorithm presented here is a modified version of Räcke's method that is more suitable for embedded systems while maintaining most of the advantages of Räcke's method.