

# Echtzeitfähige Taskmigration auf Many-Core Prozessoren

Peter Munk

Fachgebiet Kommunikations- und Betriebssysteme  
Technische Universität Berlin, Germany

31. März 2014

In der Vergangenheit wurde die Leistung neuer Generationen von Prozessoren hauptsächlich durch eine höhere Taktfrequenz und größere Caches verbessert. Der damit einhergehende steigende Energieverbrauch erreichte jedoch Größenordnungen, die eine weitere Steigerung der Taktfrequenz aus technologischer und ökologischer Sicht nicht mehr zweckmäßig erscheinen lässt.

Die auch weiterhin entsprechend dem Moore'schen Gesetz wachsende Anzahl der Transistoren pro Chip wird deshalb seitdem dafür genutzt, eine steigende Anzahl von Rechenkernen innerhalb eines Prozessors zu integrieren. Die Kerne können dabei in unterschiedlichen, spezialisierten Architekturen realisiert sein. Es ist davon auszugehen, dass die Zahl dieser heterogenen Kerne in Zukunft weiter steigen wird. So werden im Jahr 2020 bereits Prozessoren mit über 1000 Kernen erwartet [1].

Diese sogenannten Many-Core Prozessoren bestehen aus heterogenen Rechenkernen und verfügen über einen verteilten, Kern-lokalen Speicher. Die Verbindung zwischen den Kernen wird durch einen Kommunikationsmechanismus, z. B. ein Network-on-Chip (NoC) realisiert. Ein NoC überträgt den Aufbau und die Lösungen klassischer Netzwerktechnik auf einen Chip und stellt so Skalierbarkeit sicher.

Die Migrieren von Tasks zwischen diesen Kernen ist aus verschiedenen Gründen hilfreich. Neben den klassischen Motivationen wie Lastverteilung und Ressourcennähe kann durch Migration die Temperaturverteilung und die Leitungsaufnahme eines Many-Core Prozessors verbessert werden. Dies sorgt mittelfristig für eine geringere Fehlerwahrscheinlichkeit. Sollte ein Rechenkern dennoch ausfallen, kann mithilfe von Taskmigration die Fehlertoleranz des Systems erhöht werden. Dies ist besonders in sicherheitskritischen eingebetteten Systemen von hoher Priorität. Solche Systeme werden häufig im echtzeitkritischen Umfeld, z.B. im Automobil eingesetzt. Deshalb muss die Taskmigration in diesen Systemen ebenfalls echtzeitfähig sein.

In diesem Vortrag wird das Problemfeld der echtzeitfähigen Migration aufgezeigt. Des Weiteren wird ein generelles Konzept zur echtzeitfähigen Migration vorgestellt. Teile dieses Konzepts sind bereits auf einem zyklenakkurat simulierten Many-Core Prozessor umgesetzt und werden im Vortrag detailliert beschrieben.

## **Literatur**

- [1] The International Technology Roadmap for Semiconductors, "System drivers," 2011. [Online]. Available: <http://www.itrs.net/Links/2011ITRS/2011Chapters/2011SysDrivers.pdf>