NoC-based CSP Support for a Java Chip Multiprocessor

Flavius Gruian
Lund University, Sweden
flavius.gruian@cs.lth.se

Martin Schoeberl
Technical University of Denmark
masca@imm.dtu.dk

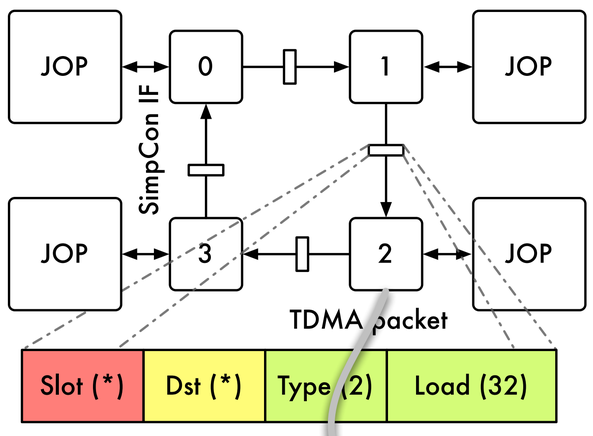
Background

Problem:
- Today only multiprocessors offer enough performance
- Shared memory scales poorly (limited bandwidth/cache coherence issues)

A Solution:
- Hoare's Communicating Sequential Processes (CSP) [1]- (Transputers/Occam)

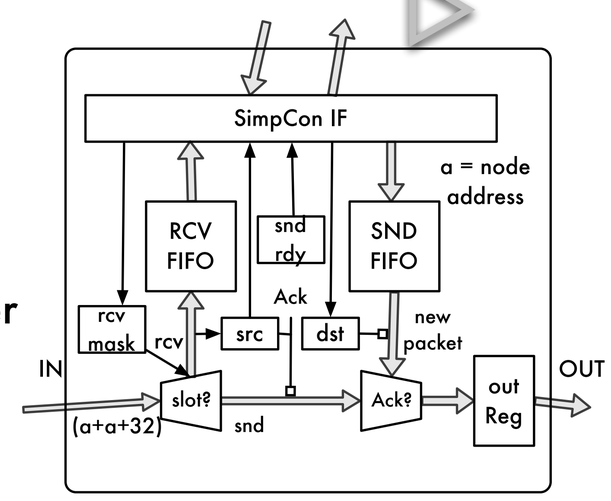
Hardware

Chip multiprocessor:
- On FPGA
- Java Optimized Processors (JOP) [2]
- A predictable network on chip (NoC)



JOP:
- Java embedded processor, direct bytecode execution
- Predictable, suitable for real-time
- Used previously in a CMP based on shared memory

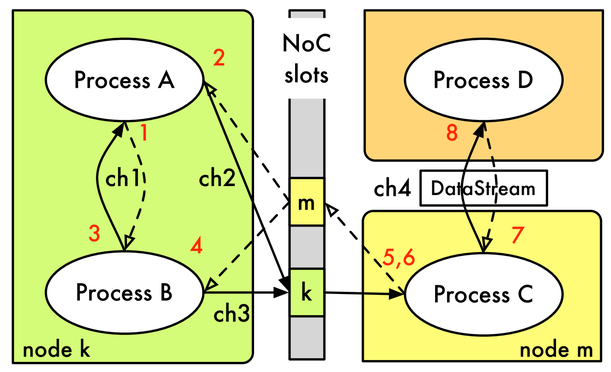
NoC:
- TDMA-based (each processor has a slot it can write data to)
- Sends packets around
- Similar to a shift register
- Ring/mesh topology
- Very simple routers



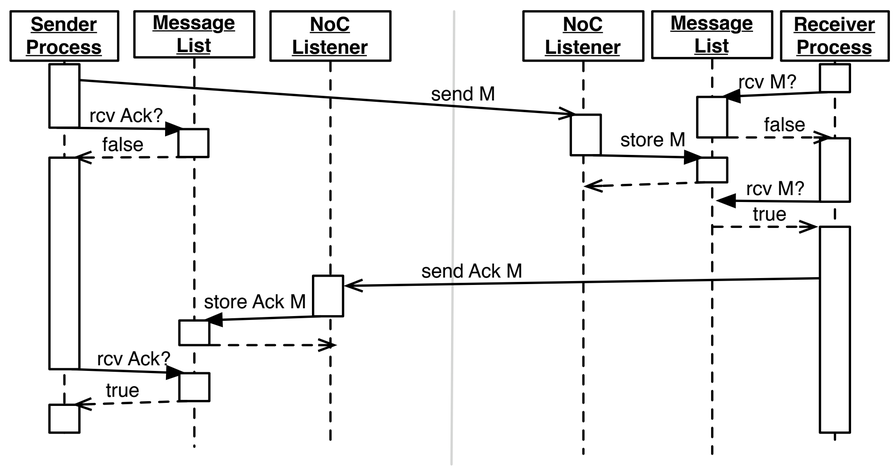
Routers:
- Can send/receive one message at the time
- Messages are made of packets
- Handle packets of types: *Nil, Data, EoD, Ack*
- Send or forward one word/clock cycle
- Reply with Ack for each Data/EoD

Software

Basics:
- Processes share processors
- CSP channels share physical channels
- CSP channels may map to local, NoC or even stream channels



Sharing NoC channels:
- Several CSP channels map to the same NoC slot
- CSP requires synchronous message passing (both the sender and the receiver block), so...
- ...use two asynchronous channels to implement this, one for the message and one for the Ack
- NoCListener: a system task to manage channels



Under development:
- Automatic mapping of processes and channels
- A more complete class library, with Occam-like ALT, PRI ALT, PAR, PRI PAR

Evaluation

Setup:
- Routers and NoC implemented in VHDL
- A system with three JOPs synthesized/tested on Altera Cyclone (EP1C12) and Digilent Nexys2 (XC3S1200)
- A Java library of 11 classes supporting local, NoC, and stream channels was also implemented and tested
- Communication speed-up for shared heap (with TDMA arbitration [3]) vs. channels on the three JOP system?

Results:
- NoC adds 15% to the design resource consumption
- On *Altera Cyclone* communication via NoC is 2.3x (5.1x) faster than shared SRAM for short (long) packets
- On *Digilent Nexys2* communication via NoC is 3.8x (11.5x) faster than shared onboard SRAM for short (long) packets

[1] C. A. R. Hoare. Communicating sequential processes. Commun. ACM, 21(8):666–677, 1978.
[2] M. Schoeberl. A Java processor architecture for embedded real-time systems. Journal of Systems Architecture, 54/1–2:265–286, 2008.
[3] C. Pitter and M. Schoeberl. A real-time Java chip-multiprocessor. ACM Trans. Embed. Comput. Syst., 10(1):9:1–34, 2010

- Files access at http://www.jopwiki.com/Download
- VHDL modules in vhdl/paper/csp (Altera Cyclone) and vhdl/paper/nexys2_csp (Nexys2)
- software support for CSP in java/target/src/paper/csp