

Supporto a Meccanismi di Comunicazione per Architetture Many-Core

Candidato:
Federico Mariti

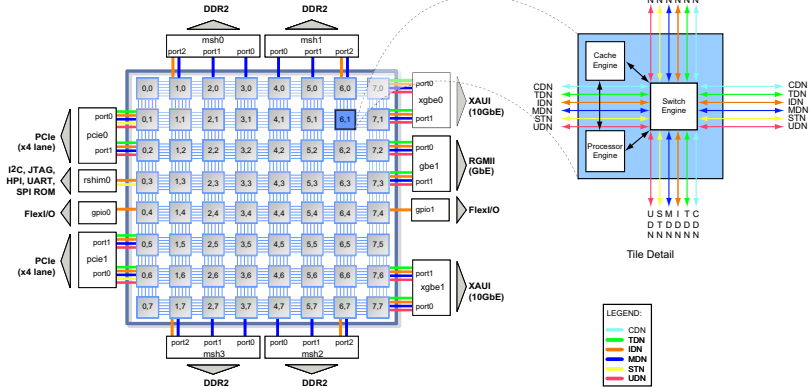
Relatore:
prof. Marco Vanneschi

21 giugno 2013

Scopo

- ▶ Prime sperimentazioni sull'uso di Network on Chip per la realizzazione di un supporto ottimizzato alle comunicazioni tra processi.
- ▶ Nuovo approccio all'implementazione del supporto.
- ▶ Lo studio è realizzato sul Chip MultiProcessor Tiler TILEPro64.
- ▶ Sono realizzate due implementazioni del supporto:
 - ▶ una che utilizza la memoria condivisa;
 - ▶ l'altra che usa la rete di interconnessione messa a disposizione dal processore.

Reti di interconnessione del Tiler TILEPro64



Dominio Applicativo

Computazioni con calcolo di grana fine:

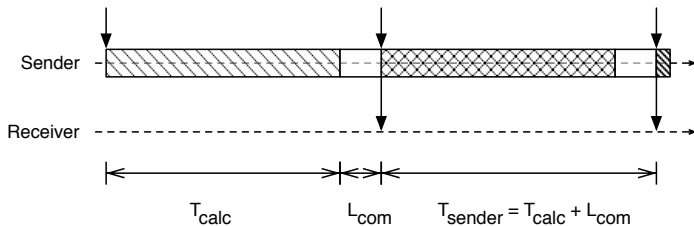
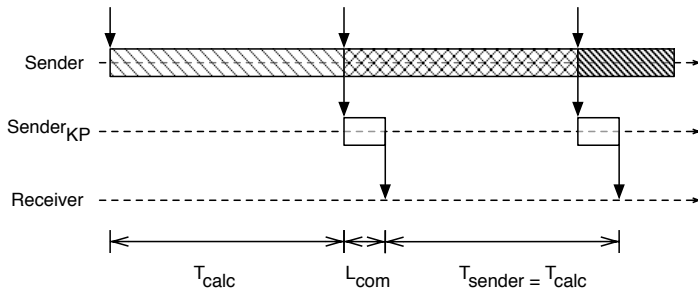
Data Stream Processing uno o più flussi contigui, rapidi e varianti nel tempo di dati; l'elaborazione sui dati in ingresso è fatta on-line:

- ▶ Monitoraggio e sicurezza di reti informatiche;
- ▶ Applicazioni finanziarie;
- ▶ Monitoraggio di sensori e Gestione delle emergenze;
- ▶ Altri ...

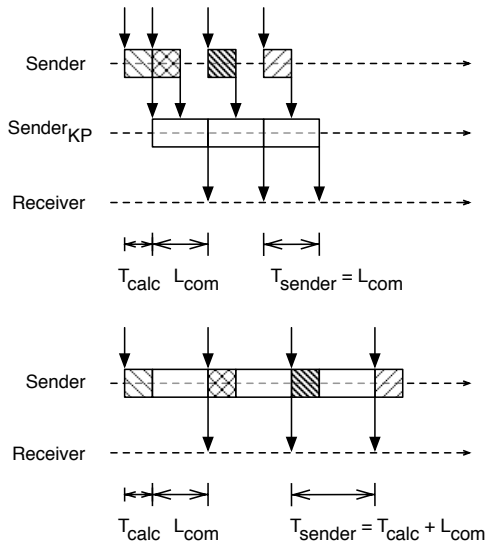
Data Parallel forma di parallelismo applicabile sia a computazioni su stream che su dato singolo; è caratterizzata dal partizionamento dei dati.

- ▶ anche su dato singolo, in particolare per la realizzazione delle comunicazioni collettive e nelle forme *Stencil*

Sulla grana del calcolo (1)

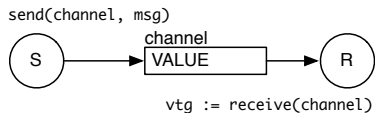


Sulla grana del calcolo (2)

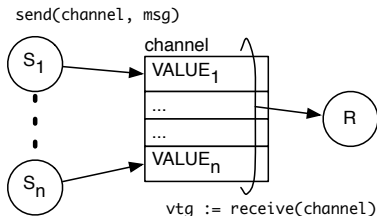


Forme di comunicazione

Canale simmetrico



Canale asimmetrico in ingresso



- ▶ Tipo 'riferimento'
- ▶ Grado asincronia unitario
- ▶ Semantica Bloccante
- ▶ Protocollo Rdy-Ack

Implementazioni del canale simmetrico

uso della Memoria Condivisa

- ▶ `ch_sym_sm_rdyack_no` gestione predefinita della coerenza della cache;
- ▶ `ch_sym_sm_rdyack` configurazione della coerenza della cache che massimizza la località delle informazioni del supporto nei core consumatori;
- ▶ `ch_sym_sm_nullack` utilizzo di un diverso protocollo di comunicazione che garantisce la correttezza senza l'uso di istruzioni di barriera di memoria

uso della UDN

- ▶ `ch_sym_udn` corrispondenza biunivoca tra i flussi firmware e i canali a livello processi

Implementazioni del canale simmetrico

uso della Memoria Condivisa

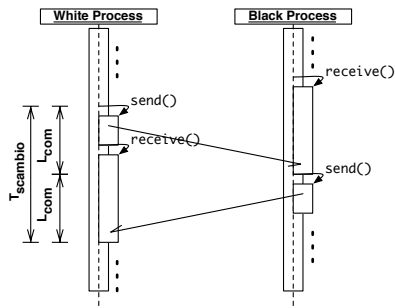
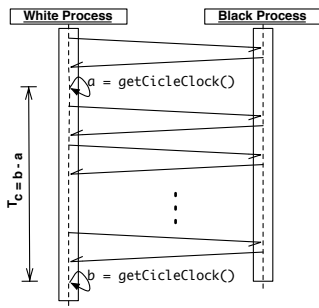
- ▶ `ch_asymin_sm_all`
- ▶ `ch_asymin_sm`

uso della UDN

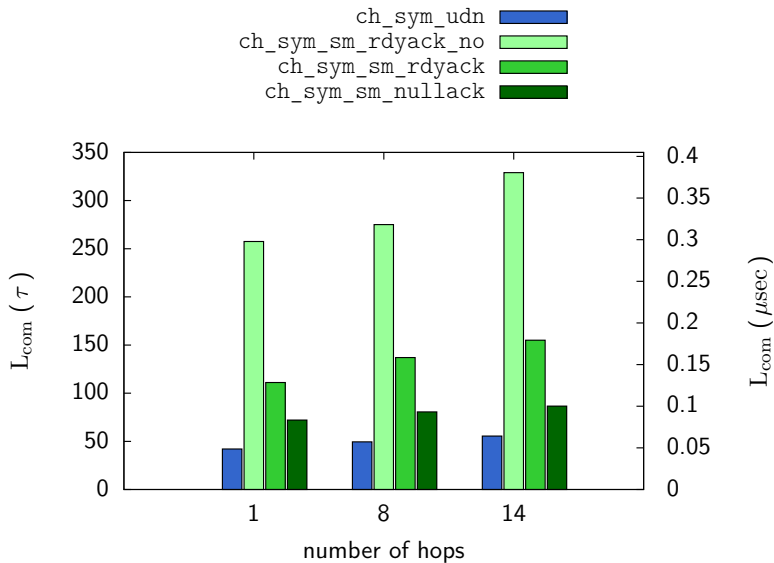
- ▶ `ch_asymin_udn`

Misura della latenza di comunicazione

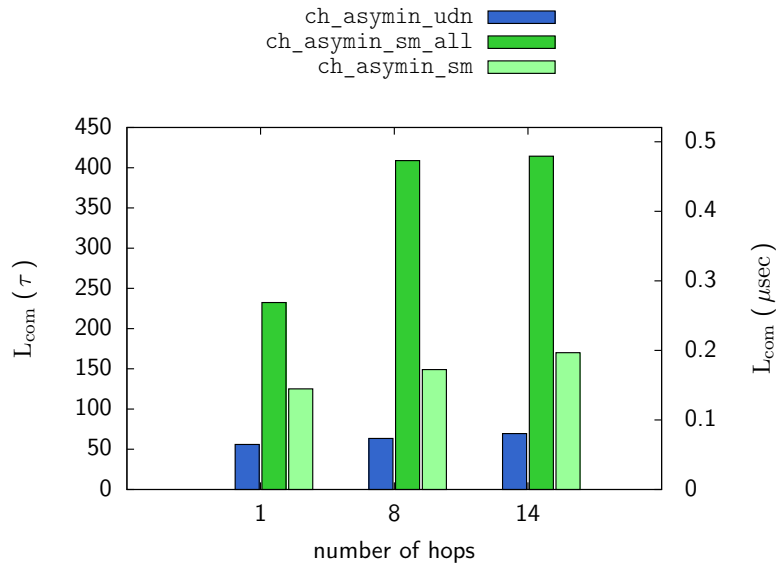
- ▶ La latenza di comunicazione è misurata per mezzo di una applicazione “ping-pong”:
 - ▶ composta da due processi collegati da due canali,
 - ▶ viene svolto lo scambio di m messaggi tra i due processi;
- ▶ La latenza di comunicazione è stimata con $L_{\text{com}} = T_C / (2 \cdot m)$.



Misura della latenza del canale simmetrico



Misura della latenza del canale asimmetrico



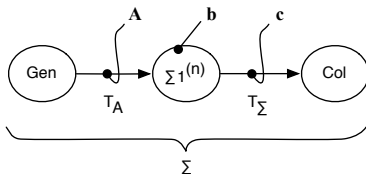
Benchmark: prodotto matrice-vettore su stream

$$\mathbf{A} = (a_{ij})_{i=1,\dots,M,j=1,\dots,M} \in \mathbb{Z}^{M \times M}$$

$$\mathbf{b} = (b_1, \dots, b_M) \in \mathbb{Z}^M$$

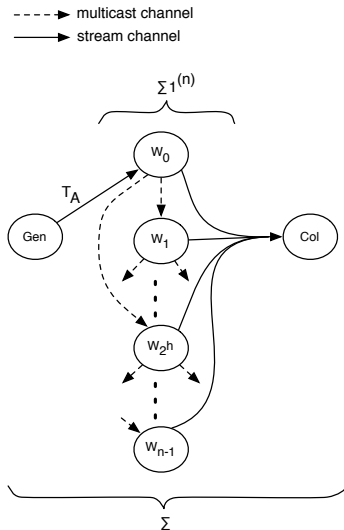
$$\mathbf{c} = (c_1, \dots, c_M) = \mathbf{A} \cdot \mathbf{b} \in \mathbb{Z}^M$$

$$\forall i \in \{1, \dots, M\} : c_i = \mathbf{a}_i \cdot \mathbf{b} = \sum_{j=1}^M a_{ij} \cdot b_j$$



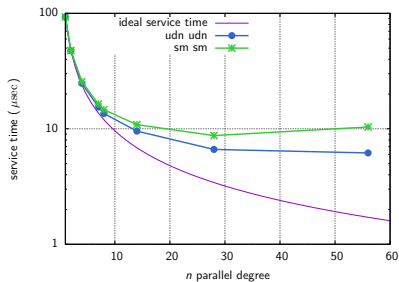
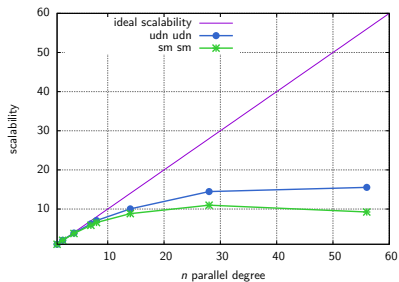
Benchmark: soluzione parallela

- Data Parallel Map
- Multicast strutturata ad albero distribuito nei worker



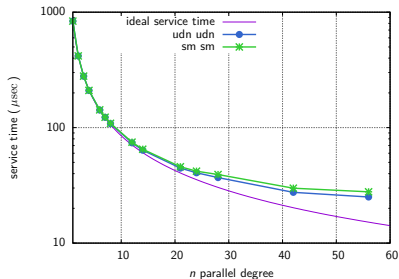
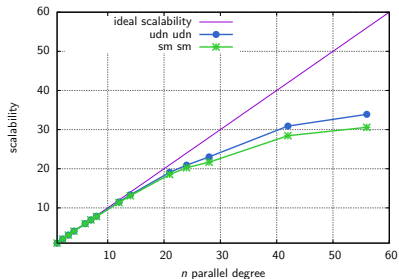
Confronto: tempo di servizio (1)

- ▶ Tempo di interarrivo $4.627 \mu\text{sec}$
- ▶ Dimensione delle matrici 56×56
- ▶ Tempo di servizio migliore UDN ... Tempo di servizio migliore SM ...



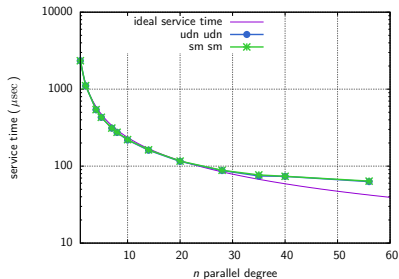
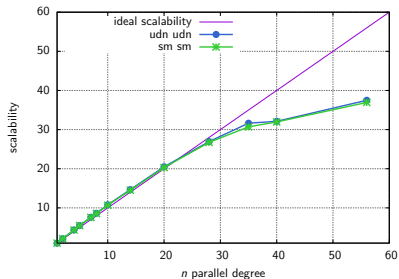
Confronto: tempo di servizio (2)

- ▶ Tempo di interarrivo $23.133 \mu\text{sec}$
- ▶ Dimensione delle matrici 168×168



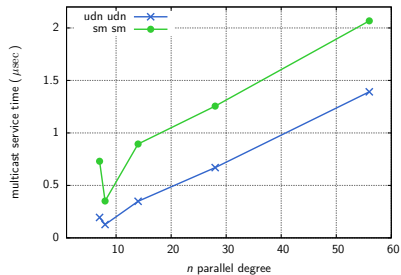
Confronto: tempo di servizio (3)

- ▶ Tempo di interarrivo $57.831 \mu\text{sec}$
- ▶ Dimensione delle matrici 280×280

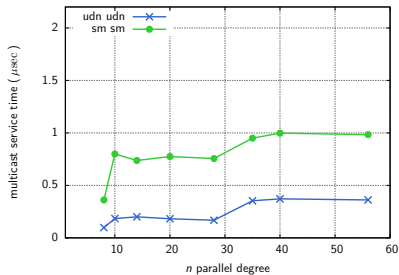


Confronto: tempo di servizio Multicast

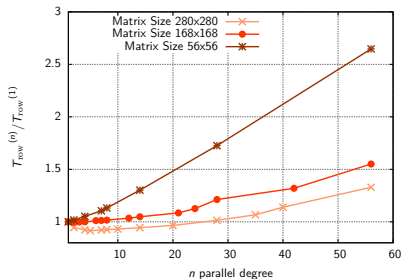
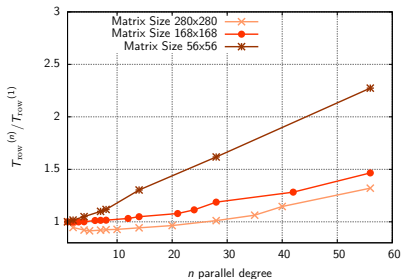
Dimensione delle matrici 56x56



Dimensione delle matrici 280x280



Confronto: tempo di calcolo di un singolo prodotto scalare



Confronto: tempo di calcolo di un singolo prodotto scalare

