



Sistemi embedded sviluppati tramite EDK: implementazione ed analisi delle prestazioni del meccanismo di DMA sul bus OPB

Relatore: Prof. Francesco Bruschi

Correlatore: Ing. Marco Domenico Santambrogio

Elaborato di laurea di:

Paolo Somaglia

Riccardo Tornese

Sommario

- Scopo del lavoro
- Prerequisiti sul DMA
- Architetture: ambiente e scelte effettuate
- Architetture: scenari implementati
- Risultati
- Conclusioni
- Sviluppi futuri

Scopo del lavoro

- 1) Analizzare le prestazioni del meccanismo di DMA
- 2) Valutare alcune possibili soluzioni implementative
- 3) Offrire la possibilità di incrementare le prestazioni del processo di riconfigurazione del sistema



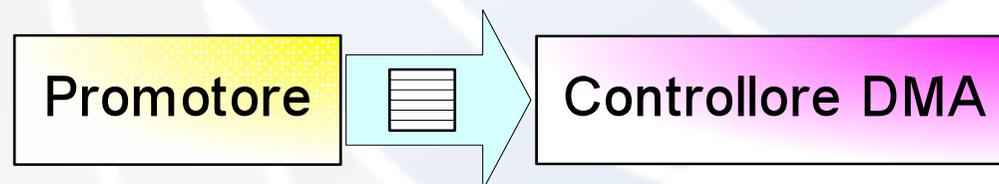
Prerequisiti sul DMA (1)

Direct Memory Access

Tecnica di trasferimento dati tra dispositivi, eseguito da un controllore dedicato invece che dal processore

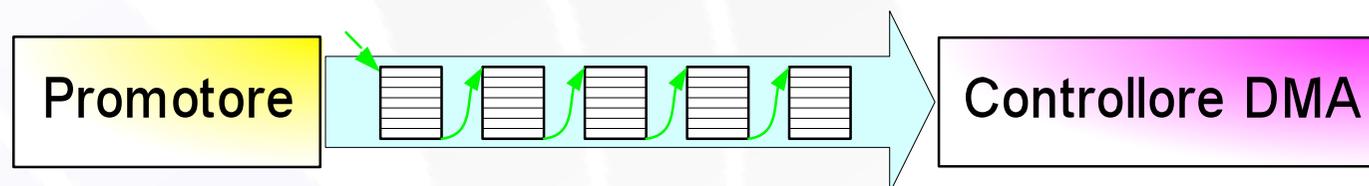
DMA semplice

Il promotore inizializza un unico trasferimento



DMA Scatter Gather (SG)

Il promotore inizializza una serie di trasferimenti

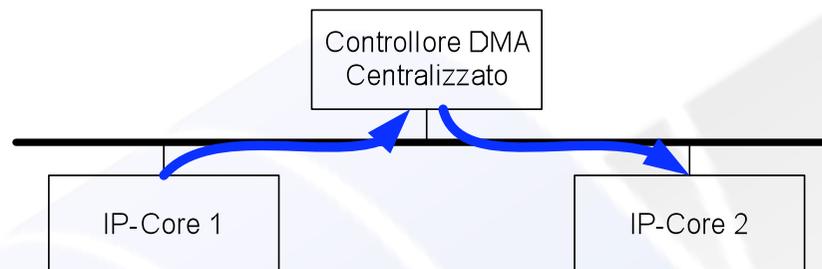


Per noi: DMA [X] DMA semplice, promotore [X] processore

Prerequisiti sul DMA (2)

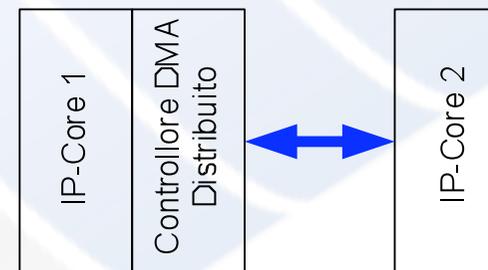
DMA Centralizzato (DMAC):

Il controllore esterno fa da ponte nel trasferimento



DMA Distribuito (DMAD):

Il controllore DMA viene inserito di una delle periferiche coinvolte nel trasferimento



Burst:

Modalità per la trasmissione di più dati senza ripetere la procedura di bus grant



Senza Burst



Con Burst

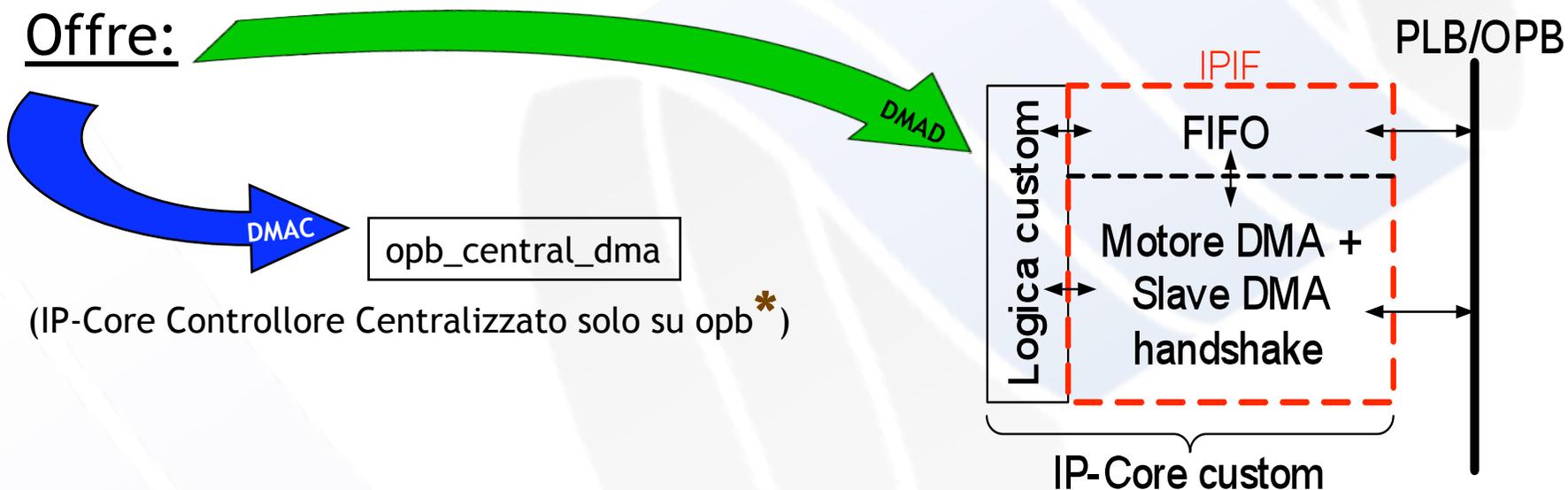
Architetture: ambiente e scelte effettuate



EDK

Embedded Development Kit
Ambiente di sviluppo di sistemi embedded

Offre:

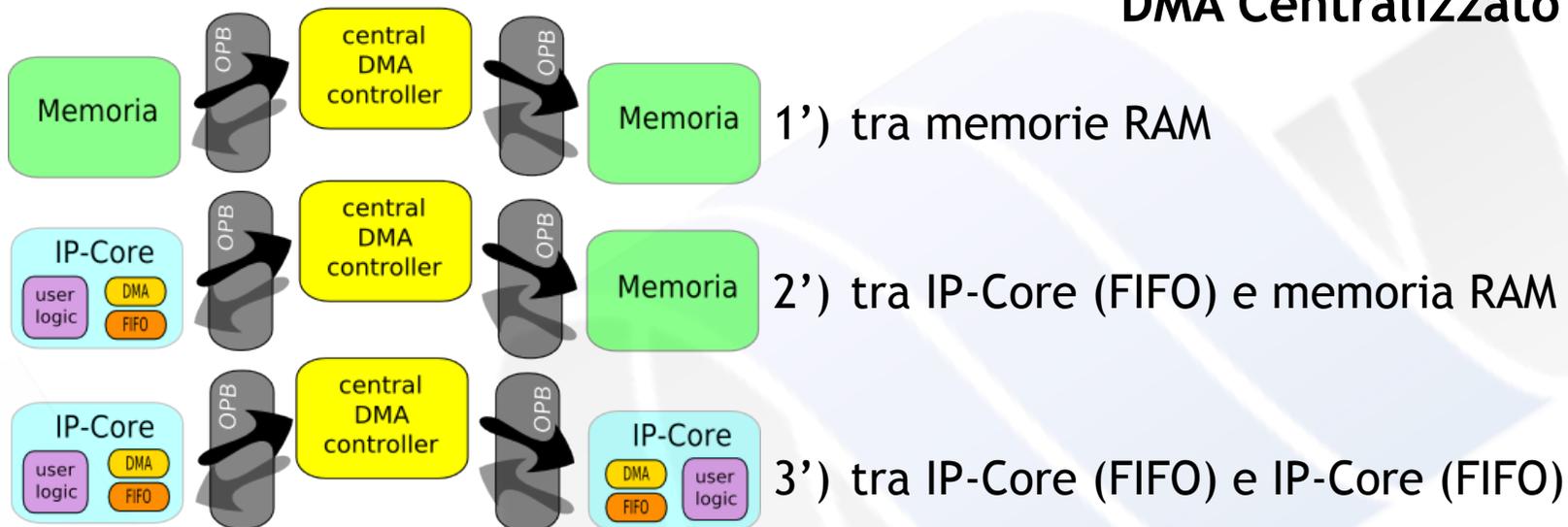


Architettura	Processore	Bus usati	Bus di DMA *
A	PowerPC™	PLB, OPB	OPB
B	MicroBlaze	OPB	OPB

Architetture: scenari implementati

Software

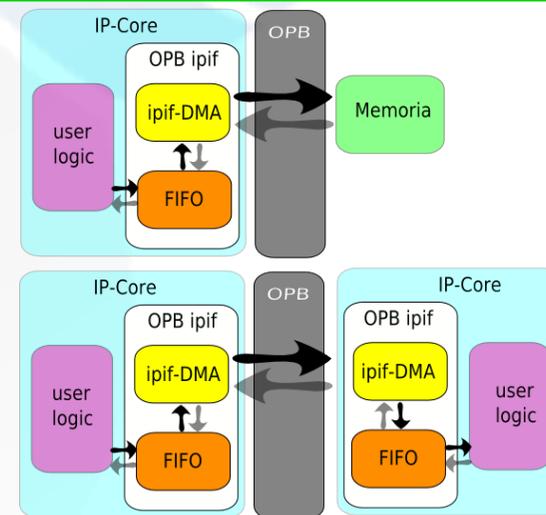
DMA Centralizzato



DMA Distribuito

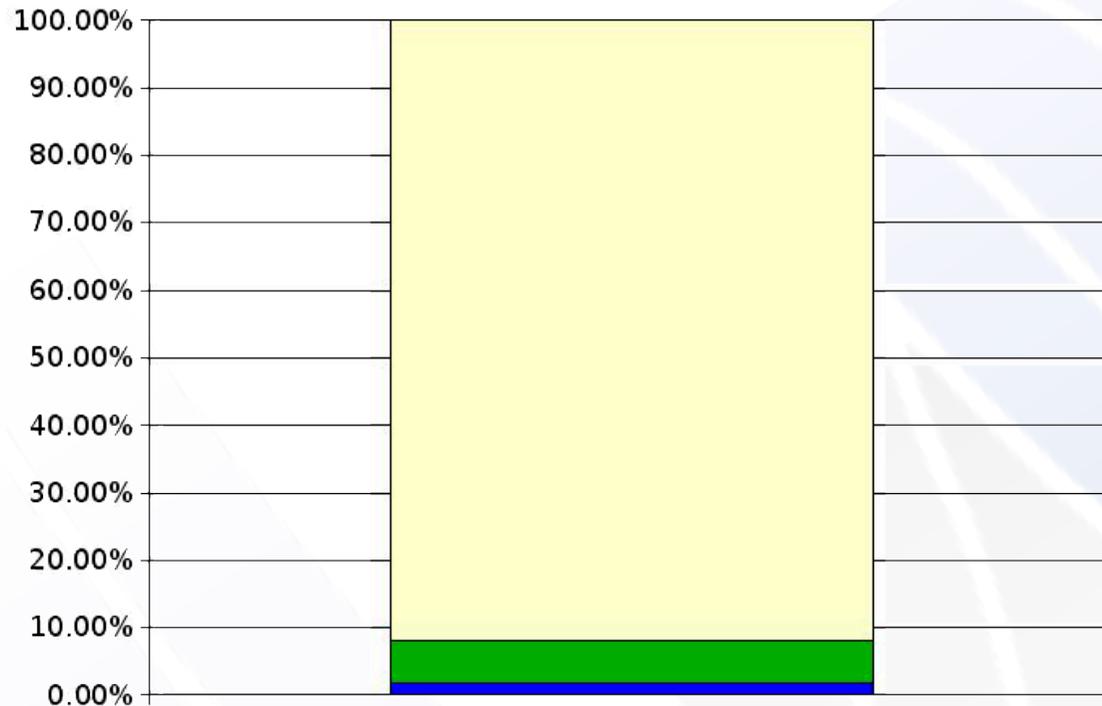
tra IP-Core (FIFO) e memoria RAM (2'')

tra IP-Core (FIFO) e IP-Core (FIFO) (3'')



Risultati (1)

Occupazione d'area

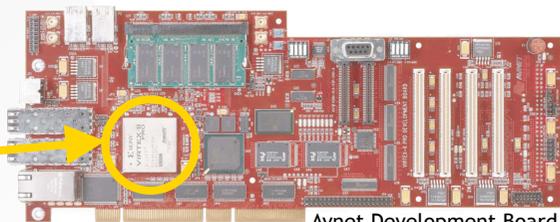


- Occupazione dell' opb_central_dma (1,93%)
- Occupazione del modulo DMA di un IP-Core (6,24%)

Dati relativi alla FPGA



XC2VP20



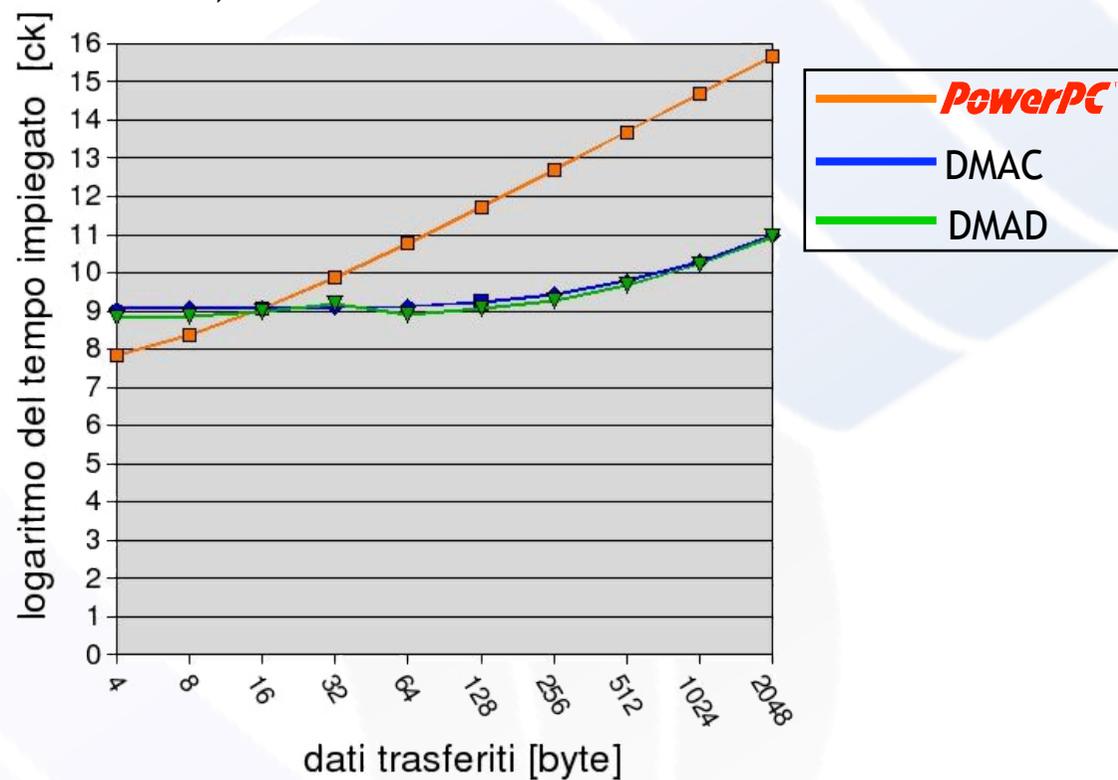
Avnet Development Board

Risultati (2)

Tempi (Architettura A)

Retta sul grafico logaritmico: $y = \log_2(\frac{t}{t_{OH}})x + t_{OH}$

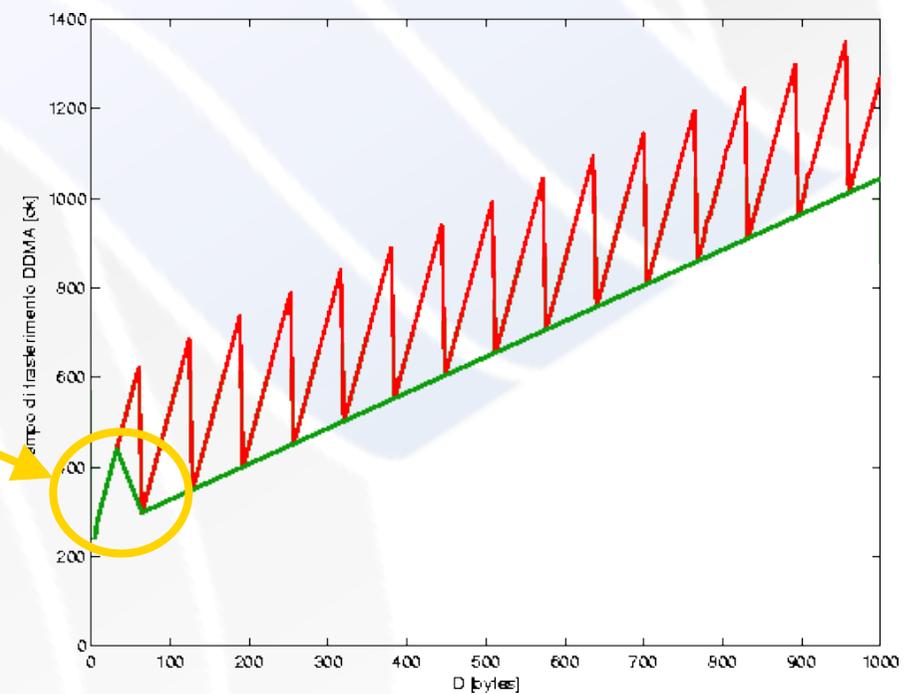
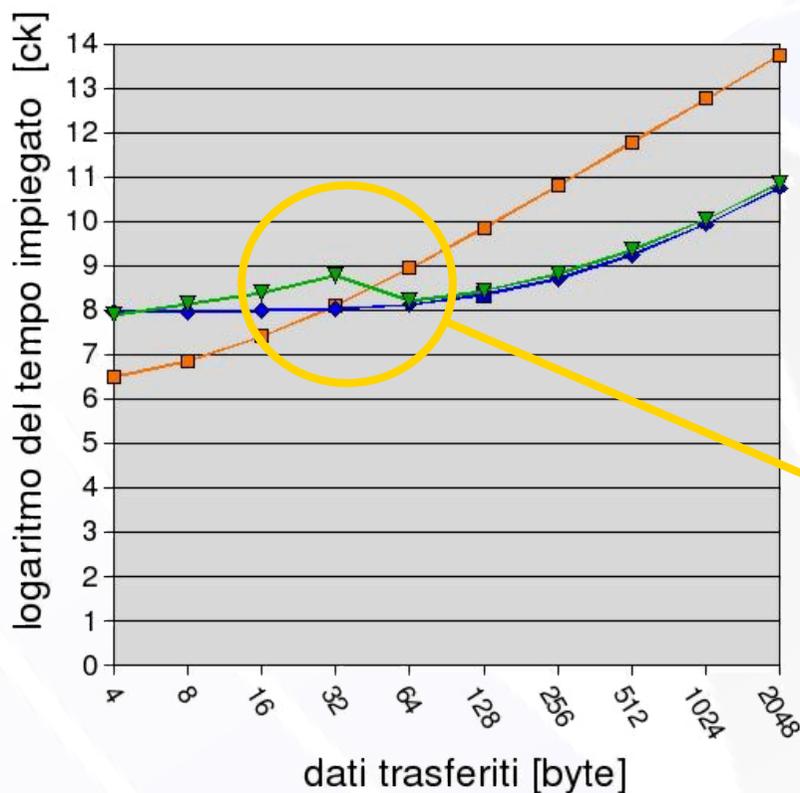
2) FIFO → SDRAM



Risultati (3)

Tempi (Architettura B)

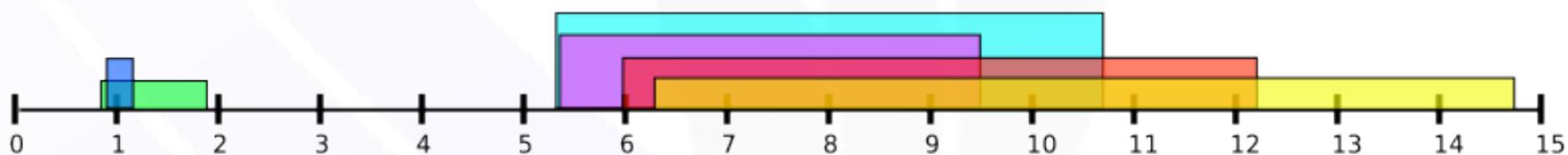
2) FIFO → SDRAM



Risultati (4)

Rapporti di prestazione

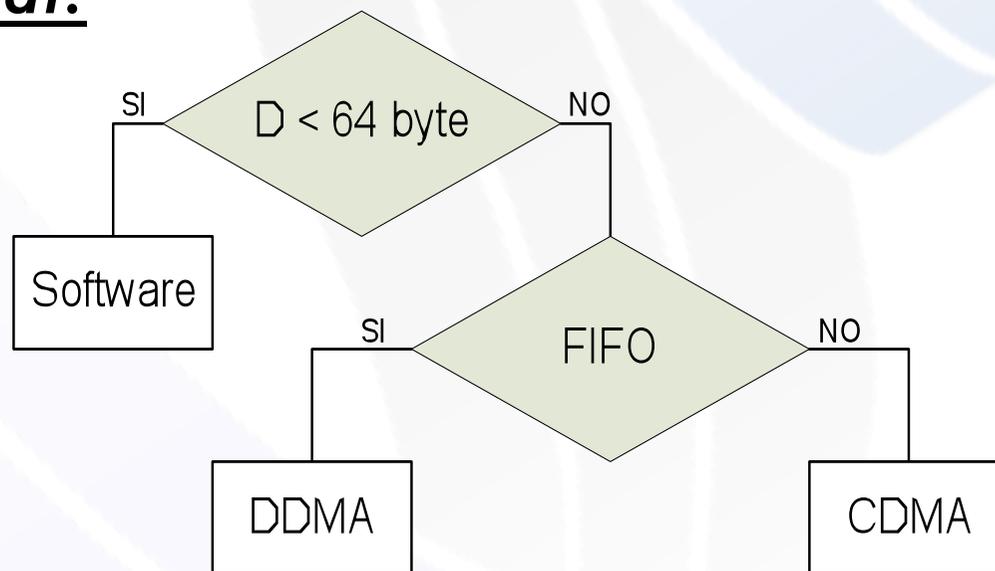
- range del rapporto fra la velocità del DMA distribuito e centralizzato
- range del rapporto fra la velocità del DMA centralizzato ed il Microblaze
- range del rapporto fra la velocità del DMA distribuito ed il Microblaze
- range del rapporto fra la velocità del DMA centralizzato ed il PowerPC
- range del rapporto fra la velocità del DMA distribuito ed il PowerPC
- range del rapporto fra la velocità del DMA distribuito senza burst ed il Microblaze



Conclusioni

- La velocità del trasferimento viene al massimo raddoppiata con DMA senza burst, viene più che quintuplicata se il burst viene attivato
- Conviene usare il processore per trasferimenti “piccoli” (< 64 byte)
- Il DMAD ha una velocità maggiore del DMAC di una quantità < 20%
- L’IP-Core con FIFO necessita del modulo DMA nell’IPIF

...Quindi:



Sviluppi futuri

- Implementazione ed analisi delle prestazioni del DMA:
 - ✓ in modalità scatter gather
 - ✓ su bus PLB
 - ✓ in esempi applicativi
- Introduzione del DMA nei lavori di tesi che lo prevedono come possibile sviluppo futuro

Fine presentazione

Domande?