

# Hard real-time meets embedded multicore NIOS SoPCs

**Primiano Tucci**

Università di Bologna

DEIS – Dipartimento di Elettronica, Informatica e Sistemistica



## Il paradigma System on (Programmable) Chip come fattore abilitativo per le tecnologie Real-Time embedded

- Consolidamento della infrastruttura hardware, integrando:
  - Piattaforme computazionali general-purpose (NIOS-II soft-processor)
  - Hardware custom special-purpose
  - Acceleratori / IP Core di terze parti (FFT, AES, MAC ...)
- Possibilità di realizzare configurazioni multi-processore

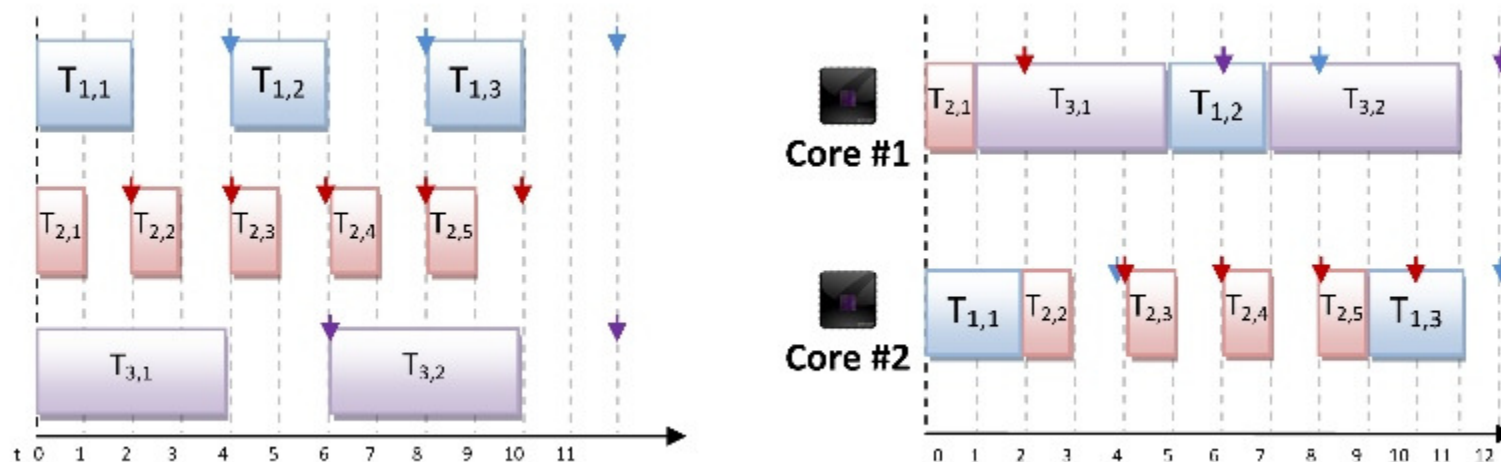


**Nuove sfide sia per quanto riguarda sia il software che l'hardware.**

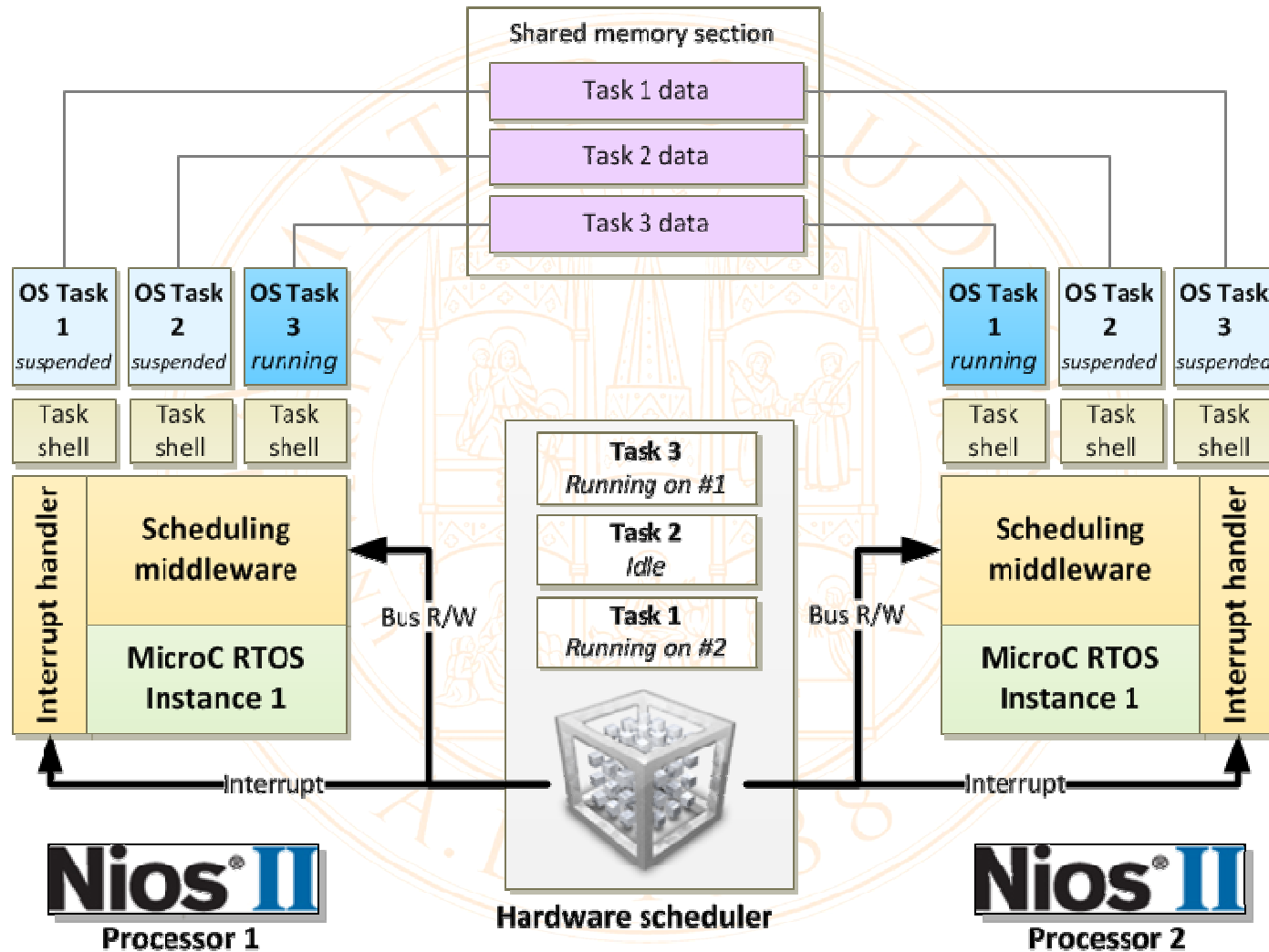
# Obiettivo del progetto

## Piattaforma real-time multi-processore

- Definizione della architettura hardware
- Definizione della architettura software
- Design dei componenti necessari alla integrazione HW/SW



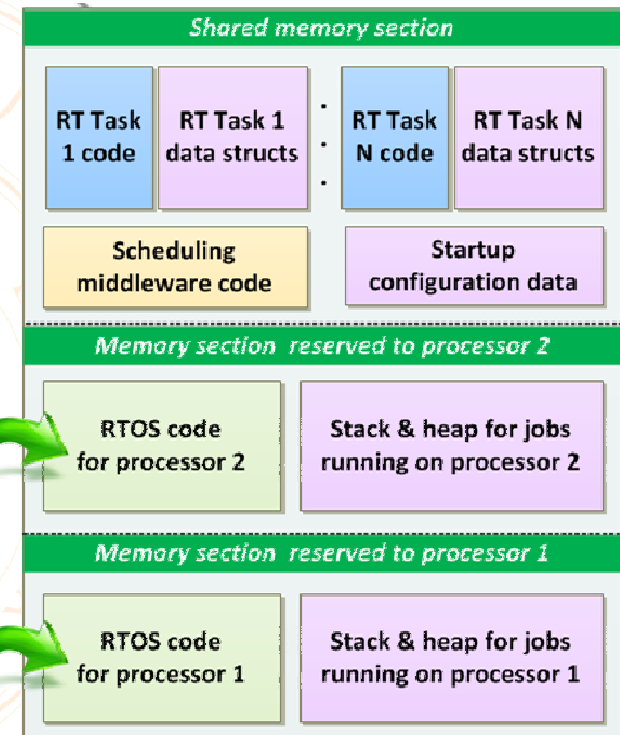
# Architettura software



# Layout software

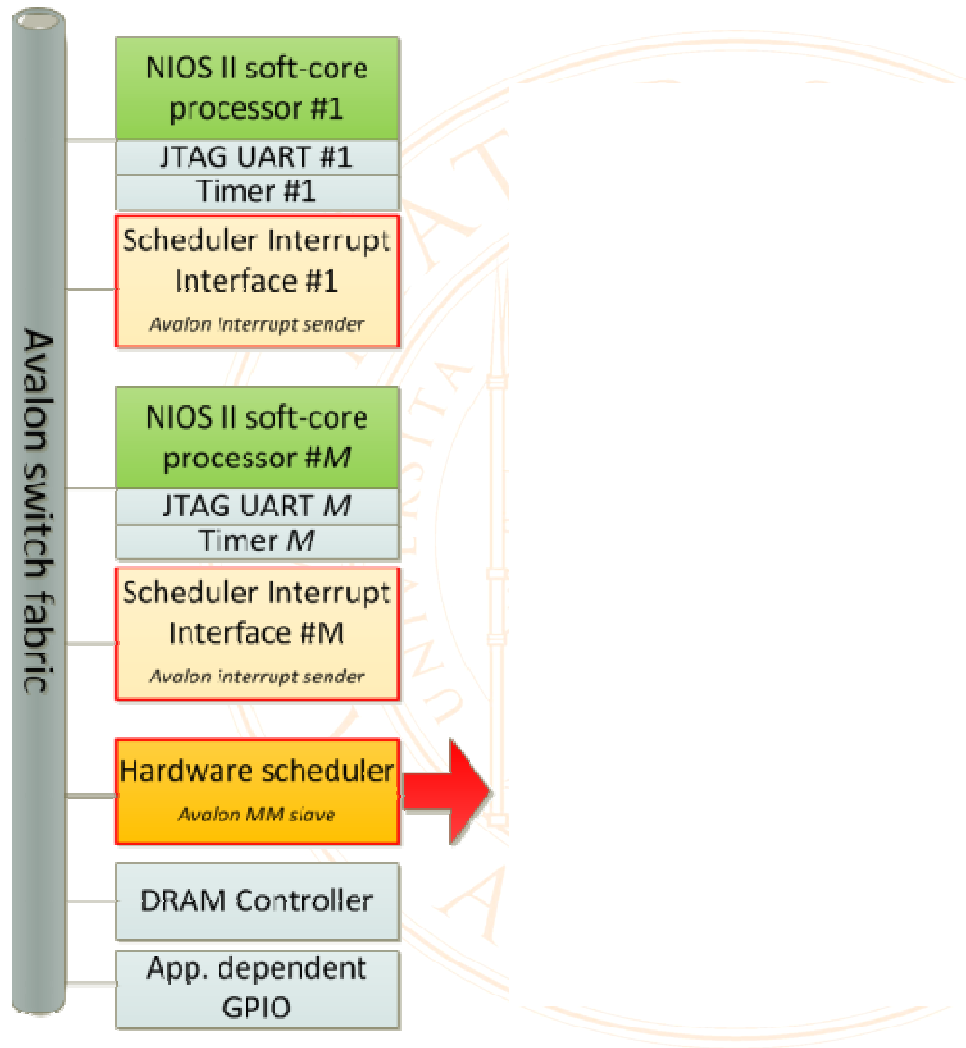
## Problematiche

- Inapplicabilità dell'approccio SMP (coerenza cache, meccanismi IPC, maggiore complessità software)
- Approccio AMP, istanze indipendenti del RTOS su ogni processore.
- Coordinare dinamicamente l'avanzamento dei task tra i vari RTOS, garantendo il rispetto dei vincoli temporali
- Intervenire in modo *portabile* senza modificare il sistema operativo ma sfruttando le primitive da esso messe a disposizione.





# Architettura hardware





# Integrazione nella piattaforma SoPC

Altera SOPC Builder - NIOSSystem.sopc (D:\Progetti\RTNIOSystem.sopc)

System Contents | System Generation

Component Library

Project: New component... Library

- Accelerators
  - Multicore Scheduler
  - Scheduler Interrupt Interface
- Avalon Verification Suite
- Bridges and Adapters
- Interface Protocols
- Legacy components
- Memories and Memory Controllers
- Peripherals
- PLL
- Processor Additions
- Processors
- SLS
- Terasic Technologies Inc
- University Program
- Video and Image Processing

Target: Device Family: Cyclone II

Clock Settings

Name	Source	MHz	Add	Remove
clk_0	External	50,0		

Use Connections Module Name Description Clock IRQ

- SW PIO (Parallel IO)
- s1 Avalon Memory Mapped Slave clk\_0
- dram SDRAM Controller
- s1 Avalon Memory Mapped Slave clk\_0
- mutex\_0 Mutex
- s1 Avalon Memory Mapped Slave clk\_0
- performance\_counter Performance Counter Unit
- control\_slave Avalon Memory Mapped Slave clk\_0
- BUTTONS PIO (Parallel IO)
- s1 Avalon Memory Mapped Slave clk\_0
- HEX1 PIO (Parallel IO)
- s1 Avalon Memory Mapped Slave clk\_0
- HEX2 PIO (Parallel IO)
- s1 Avalon Memory Mapped Slave clk\_0
- HEX3 PIO (Parallel IO)
- s1 Avalon Memory Mapped Slave clk\_0
- HEX4 PIO (Parallel IO)
- s1 Avalon Memory Mapped Slave clk\_0
- MOTOR\_1 Avalon Memory Mapped Slave clk\_0
- s1 PIO (Parallel IO)
- MOTOR\_2 Avalon Memory Mapped Slave clk\_0
- s1 PIO (Parallel IO)
- Multicore Scheduler\_0 Multicore Scheduler
- scheduler\_slave Avalon Memory Mapped Slave clk\_0
- cpu\_1.instruction\_master Nios II Processor
- data\_master Avalon Memory Mapped Master
- jtag\_debug\_module Avalon Memory Mapped Slave
- jtag\_uart\_0 JTAG UART
- avalon\_ftag\_slave Avalon Memory Mapped Slave clk\_0
- timer\_cpu0 Interval Timer
- s1 Avalon Memory Mapped Slave clk\_0
- SchedulerInterruptInterface\_0 Scheduler Interrupt Interface
- scheduler\_int\_slave Avalon Memory Mapped Slave clk\_0
- rs232 UART (RS-232 Serial Port)
- s1 Avalon Memory Mapped Slave clk\_0
- cpu\_1 Nios II Processor
- instruction\_master Avalon Memory Mapped Master clk\_0
- data\_master Avalon Memory Mapped Master
- jtag\_debug\_module Avalon Memory Mapped Slave
- jtag\_uart\_1 JTAG UART
- avalon\_ftag\_slave Avalon Memory Mapped Slave clk\_0
- timer\_cpu1 Interval Timer
- s1 Avalon Memory Mapped Slave clk\_0
- SchedulerInterruptInterface\_1 Scheduler Interrupt Interface
- scheduler\_int\_slave Avalon Memory Mapped Slave clk\_0

Info: BUTTONS: PIO inputs are not hardwired in test bench. Undefined values will be read from PIO inputs during simulation.

Generate

```
void SharedSystemStartup()
{
    ApiResponse res;
    printf("Welcome to HWScheulder on CPU %d\n",
        get_cpuid());

    hwsched_uc_init();

    res = hwsched_uc_create_periodic_task(
        20, /* Task period */
        20, /* Task deadline */
        motor_control, /* Job entry point */
        &motor_conrol_data /* Job data structures */
    );

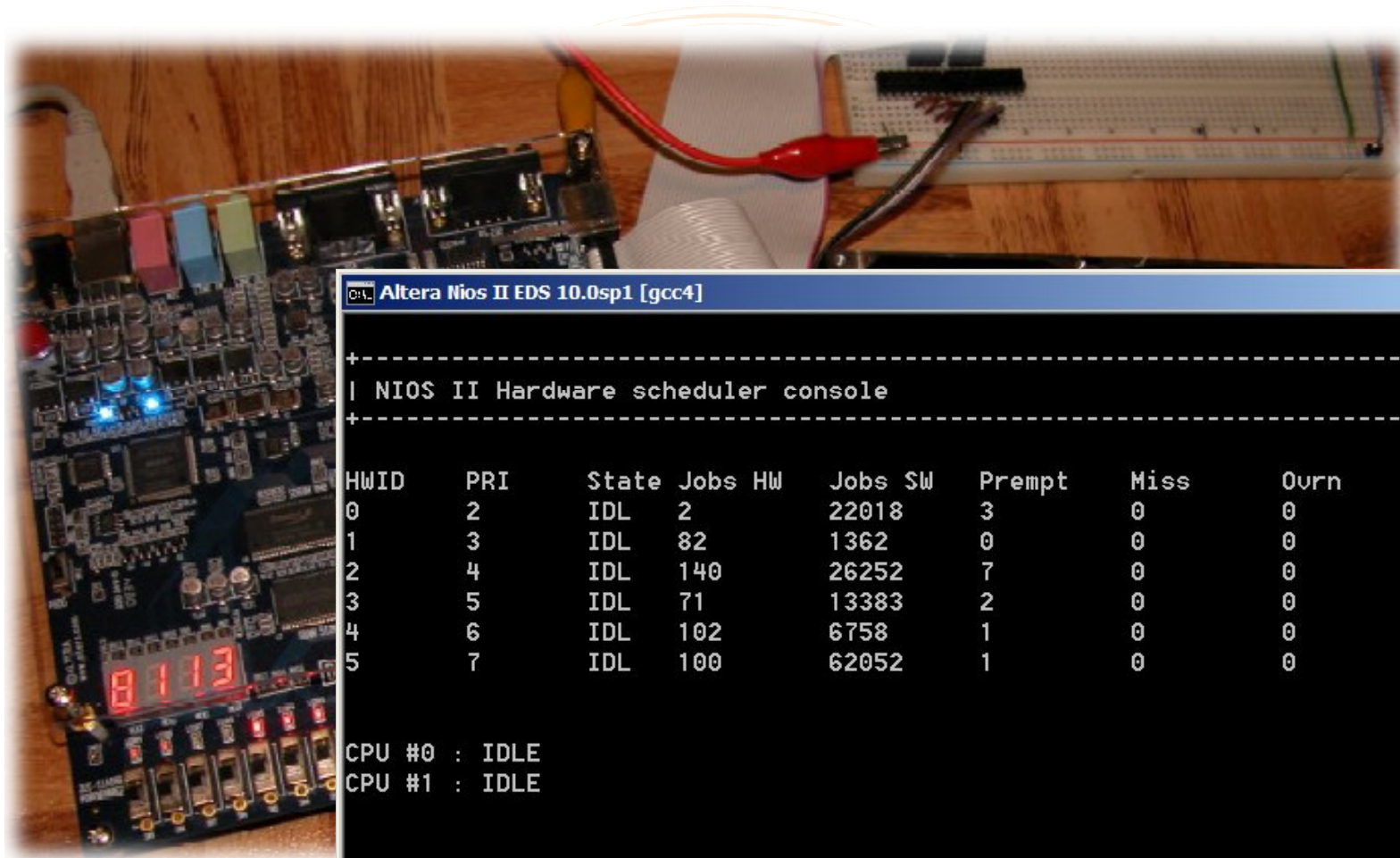
    printf("Motor control task created, res = %d\n",res);

    /* Repeat the same for each real-time task */

    printf("Entering real-time mode\n");
    res = hwsched_uc_start();
    if (res != S_OK)
    {
        printf("Error during scheduler initialization\n");
        for(;;) {}
    }
}
```



# Implementazione



```
Altera Nios II EDS 10.0sp1 [gcc4]
-----+-----+
| NIOS II Hardware scheduler console |
+-----+-----+
HWID   PRI   State Jobs HW   Jobs SW   Preempt   Miss   Ourn
0      2     IDL   2    22018   3         0         0
1      3     IDL   82    1362    0         0         0
2      4     IDL  140   26252   7         0         0
3      5     IDL   71    13383   2         0         0
4      6     IDL  102    6758    1         0         0
5      7     IDL  100   62052   1         0         0

CPU #0 : IDLE
CPU #1 : IDLE

ISR: Min: 245, Max: 4189
Task shell: Min: 245, Max: 1249
```





# Risultati

- Offload delle funzioni di scheduling  
La politica di scheduling è completamente gestita in hardware, riducendo al minimo l'overhead computazionale richiesto ai soft-processor.

Overhead interrupt service routine		Overhead task shell	
Min	Max	Min	Max
245 T CLK (4.9 us)	4189 T CLK (83.78 us)	245 T CLK (4.9 us)	1249 T CLK (24.98 us)

- Basso jitter nel rilascio di processi periodi (10 ns @ 50 MHz)
- Adattabilità dell'hardware in base alle necessità

Configuration	Total comb. functions	Total logic registers	f <sub>MAX</sub>
2 cores, 8 Tasks, 8 bit deadline and period counters, 8 bit stats counters	1246 (7%)	598 (3%)	59.8 MHz
4 cores, 16 Tasks, 8 bit deadline and period counters, 8 bit stats counters	2363 (13%)	1114 (6%)	51.86 MHz
8 cores, 32 Tasks, 16 bit deadline and period counters, 16 bit stats counters	7167 (38%)	3738 (20%)	39.08 MHz

*Synthesis on a Cyclone II EP2C20F484C7 using the Quartus II EDA (Opt.: balanced) and the TimeQuest timing analyzer (slow-model)*



ALMA MATER STUDIORUM – UNIVERSITY OF BOLOGNA

DEIS - DEPARTMENT OF ELECTRONICS, COMPUTER ENGINEERING AND SYSTEMS

**Grazie per l'attenzione**

