



[Schema della lezione](#)

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)

Lezione 31

Conclusione del corso

Sistemi embedded e real-time

31 gennaio 2013

Marco Cesati

Dipartimento di Ingegneria Civile e Ingegneria Informatica
Università degli Studi di Roma Tor Vergata

Di cosa parliamo in questa lezione?



Schema della lezione

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)

Conclusione del corso:

- 1 Sintesi degli argomenti
- 2 Il caso Mars Pathfinder
- 3 Tesi disponibili
- 4 Varie ed eventuali

I due temi principali di questo corso:

- Sistemi real-time:
 - Modellare i requisiti temporali di un sistema real-time
 - Utilizzare algoritmi di schedulazione basati su parametri temporali oggettivi
 - Verificare che nel sistema real-time tutti i task completeranno sempre entro la rispettiva scadenza
- Sistemi embedded:
 - Integrare l'hardware con il software
 - Scegliere le tecnologie più adatte
 - Scrivere software che si interfaccia direttamente con l'hardware

Tutto questo è realmente utile in pratica?



[Schema della lezione](#)

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)



[Schema della lezione](#)

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)

- Preparazione della missione:
 - iniziata nel 1994
 - durata circa 3 anni
- Costo:
 - Circa 150 milioni di dollari per la sonda
 - Circa 280 milioni di dollari in totale (incluso razzo lanciatore, controllo a terra. . .)
- Tecnologie innovative utilizzate:
 - Discesa rallentata da paracadute, razzi e enormi “air-bag”
 - Veicolo semovente per analisi scientifiche (*rover*)
 - Aggiramento automatico degli ostacoli sul percorso del rover
- Razzo lanciato il 4 dicembre 1996
- Sonda atterrata il 4 luglio 1997
- Durata programmata della missione: da 7 a 30 giorni marziani

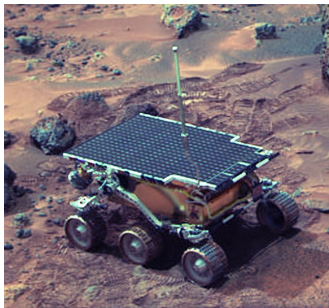
La missione Mars Pathfinder (2)



Fonte: NASA (Public Domain)

La sonda spaziale della NASA **Mars Pathfinder** è “atterrata” su Marte il 4 luglio 1997

La missione è stata definita fin da subito un “successo”: poche ore dopo l’atterraggio è stato sbarcato il veicolo rover chiamato **Sojourner**



Fonte: NASA (Public Domain)



Houston, we've had a problem...

La sonda iniziò la sua attività spedendo immagini sulla terra: tutto sembrava funzionare per il meglio

Qualche giorno dopo l'atterraggio la sonda cominciò a raccogliere e spedire dati meteorologici

Contemporaneamente, i tecnici della NASA cominciarono ad accorgersi che qualcosa non andava: la stampa parlò di “[software glitches](#)”

In effetti la sonda periodicamente subiva un [reset](#) totale che rendeva impossibile la raccolta e trasmissione dei dati

Fortunatamente i tecnici riuscirono a capire l'origine del problema e a porvi rimedio modificando a distanza il codice del programma della sonda: la missione era salva!

Ma cosa avvenne realmente?



[Schema della lezione](#)

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)



[Schema della lezione](#)

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)

- Una singola CPU RS/6000 controllava la sonda
- La CPU era collegata con un bus VME a radio, fotocamera ed una interfaccia ad un bus 1553
- Sul bus 1553 vi erano molti dispositivi utilizzati per la astro-navigazione e per l'atterraggio, nonché uno strumento per le analisi meteorologiche (ASI/MET)
- Il sistema operativo real-time era una versione di VxWorks
- Il programma per controllare il bus 1553 ed i relativi dispositivi era implementato come due task periodici ad alta priorità, ciascuno con frequenza 8 Hz:
 - il 1° task ("[bc_sched](#)") controllava le transazioni del bus
 - il 2° task ("[bc_dist](#)") gestiva i risultati, ossia i dati
 - ciascuno dei due task all'attivazione controllava che il job precedente dell'altro task fosse terminato; in caso contrario, forzava un reset di sistema
- un task di bassa priorità ("[asi_met](#)") per la raccolta dei dati meteorologici aspettava i dati dei sensori inviati da [bc_dist](#) tramite un meccanismo IPC ("pipe")
- internamente la pipe utilizzava un mutex per la sincronizzazione

Il problema di Mars Pathfinder e la sua soluzione

- Il task `bc_dist` era bloccato sul mutex acquisito da `asi_met`: **inversione di priorità!**
- Nello scenario operativo, l'insieme delle attività di raccolta dei dati attivava un gran numero di task di priorità media (intermedia tra `bc_dist` e `asi_met`)
- I task di priorità media rallentavano così tanto `asi_met` che `bc_dist` mancava la sua scadenza: **inversione di priorità non controllata!**
- I test fatti a terra non avevano consentito di scoprire il problema perché erano stati fatti con un carico di dati sul bus "nominale"
- La radice del problema: per default VxWorks non utilizzava un protocollo di **priority inheritance** sui propri mutex
- La soluzione: una variabile globale di VxWorks (non documentata!) permetteva di attivare la **priority inheritance** sui mutex
- Una patch inviata dalla Terra ha modificato la variabile e risolto il problema





[Schema della lezione](#)

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)

- Quando si usano prodotti non sviluppati in proprio, è necessario conoscerli a fondo
- Non è possibile fare *troppi* test! Più sono, meglio è
- Il problema è stato risolto perché sulla sonda era installato lo stesso codice utilizzato per sviluppo e debug sulla Terra (“test what you fly and fly what you test” — NASA)
- La teoria non è mai estranea alla pratica!

Come è andata a finire?

- L'ultima trasmissione della sonda avvenne il 27 settembre 1997, dopo 83 giorni marziani dall'atterraggio
- Non conosciamo le cause della perdita di comunicazioni: l'ipotesi più probabile è la rottura della batteria della sonda
- Nel frattempo la sonda aveva inviato 17 000 foto, oltre a 15 analisi di rocce e suolo e molti dati meteorologici



[Schema della lezione](#)

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)

- Sviluppo, integrazione e applicazioni di RTOS per sistemi embedded
 - In collaborazione con ing. D. Carnevale / S. Galeani e aziende del settore micro-elettronica
- Ricerca sulle interfacce e le applicazioni dei chip che integrano CPU e FPGA (Xilinx Zynq)
 - In collaborazione con prof. G. Cardarilli e prof. M. Re (ing. elettronica)
- Certificazione di Linux in ambito hard real-time
 - In collaborazione con industrie del gruppo Finmeccanica
- Soluzioni hardware e software per sistemi hard real-time multi-core
 - In collabor. con univ. dell'Illinois ad Urbana-Champaign (prof. M. Caccamo, prof. L. Sha)



[Schema della lezione](#)

[Sintesi](#)

[Mars Pathfinder](#)

[Tesi disponibili](#)

[Varie ed eventuali](#)

- Algoritmi di schedulazione RT per multiprocessore
 - In collaborazione con ing. A. Bastoni
- Algoritmi di schedulazione per sistemi High Performance Computing
 - In collaborazione con ing. R. Gioiosa (Barcelona Supercomputing Center)
- Implementazione ed ottimizzazione di algoritmi algebrici e numerici
 - In collaborazione con ing. E. Betti
- Acceleratori per calcolo numerico-scientifico
 - Basati su FPGA, in collaborazione con ing. P. Palana
 - Basati su GPU (CUDA, OpenCL)
- ...



Schema della lezione

Sintesi

Mars Pathfinder

Tesi disponibili

Varie ed eventuali

