

MATITA COME SUPPORTO PER SPECIFICHE ESEGUIBILI:
FORMALIZZAZIONE INTERATTIVA DEI MICROCONTROLLER A 8 BIT FREESCALE

Relatore: Dott. CLAUDIO SACERDOTI COEN

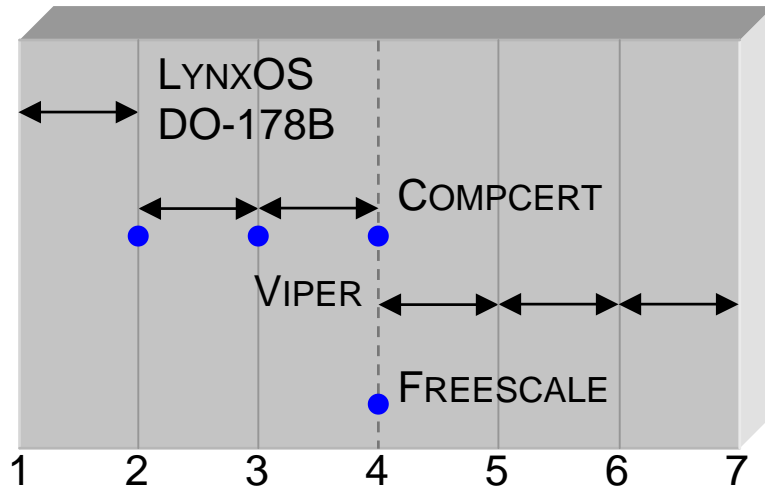
Obiettivi

- ◆ specifica eseguibile di tutti i 229 modelli di microcontroller Freescale basati su CPU a 8 bit
- ◆ presupposto per la verifica di correttezza di programmi assembler specifici per ogni modello

Ordine di Esposizione

- ◆ specifiche eseguibili
- ◆ formalizzazione
- ◆ microcontroller a 8 bit Freescale

MATITA COME SUPPORTO PER SPECIFICHE ESEGUIBILI: FORMALIZZAZIONE INTERATTIVA DEI MICROCONTROLLER A 8 BIT FREESCALE



LIVELLI DI SPECIFICA

- 1 - Sistema Operativo (SW)
- 2 - Linguaggio di Alto Livello
- 3 - Codice Intermedio
- 4 - Instruction Set (ISA) (HW)
- 5 - Stato della Macchina
- 6 - Modello a Blocchi
- 7 - Porte Logiche

● - Eseguibile

Formalizzazione

- ◆ stabilisce la corrispondenza tra specifiche di diversi livelli
- ◆ la formalizzazione CompCert (livello 2 → 4) è un compilatore

Specificazione Eseguita

- ◆ fornisce un interprete del livello di specifica
- ◆ la formalizzazione Freescale è un emulatore di microcontroller

MATITA COME SUPPORTO PER SPECIFICHE ESEGUIBILI: FORMALIZZAZIONE INTERATTIVA DEI MICROCONTROLLER A 8 BIT FREESCALE



Dimostrazione di Correttezza Hardware

- ◆ esiste un gap superiore ed inferiore
- ◆ impossibile stabilire la correttezza *oltre ogni ragionevole dubbio*

Formalizzazione Hardware

- ◆ sottopone l'hardware ad un'analisi approfondita
- ◆ strumento di razionalizzazione del design
- ◆ aumenta il grado di confidenza nel dispositivo

Gap Superiore

- ◆ forzare a compile-time la *correttezza* (A) e la *consistenza* (B)
- ◆ testare la *completezza* (C) e preservare la *semantica* (D)

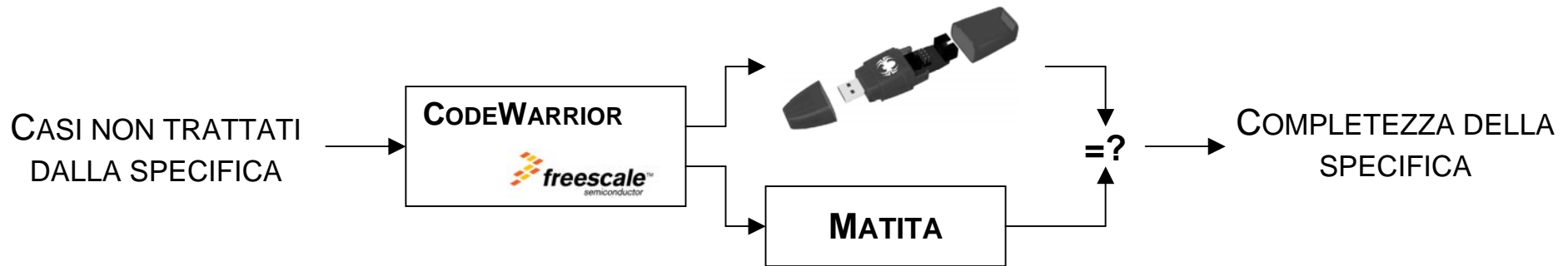
A - Correttezza della Specifica

- ◆ prevenire a compile-time molti errori di scrittura e di design
- ◆ utilizzo sistematico di tipi algebrici e dipendenti

B - Consistenza della Specifica

- ◆ prevenire a compile-time errori nelle relazioni fra le entità
- ◆ incapsulare in tipi algebrici e dipendenti le relazioni fra le entità
- ◆ verifica (dimostrazione per casi) di tutte le istanze delle relazioni

MATITA COME SUPPORTO PER SPECIFICHE ESEGUIBILI: FORMALIZZAZIONE INTERATTIVA DEI MICROCONTROLLER A 8 BIT FREESCALE



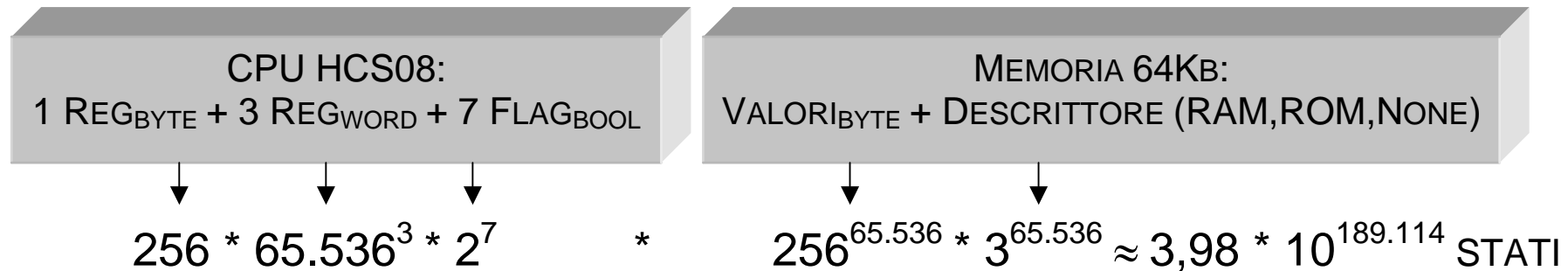
C - Completezza della Specifica

- ◆ implementare correttamente tutti i comportamenti del dispositivo
- ◆ test mirati ai casi non trattati dalla specifica

D - Semantica

- ◆ preservare la semantica di frammenti di codice reale
- ◆ test mirati a verificare il rispetto di proprietà temporali del codice (tight/upper bound) e della semantica

MATITA COME SUPPORTO PER SPECIFICHE ESEGUIBILI:
FORMALIZZAZIONE INTERATTIVA DEI MICROCONTROLLER A 8 BIT FREESCALE



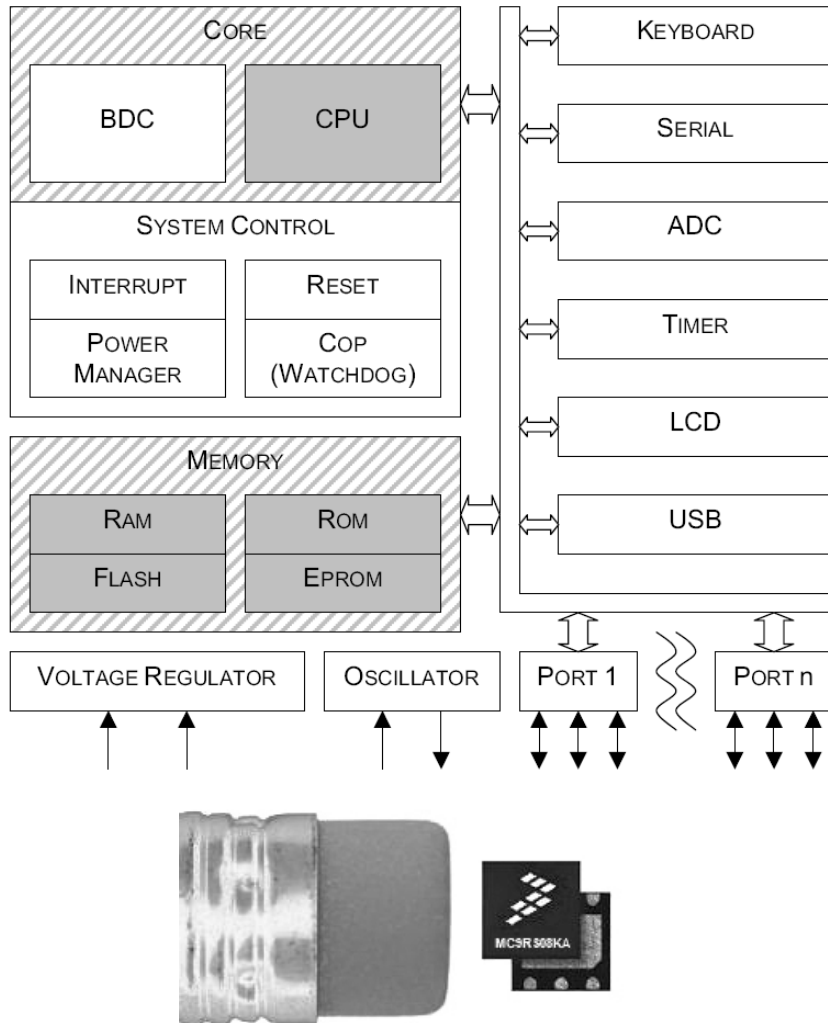
Verifica di Frammenti di Codice

- ◆ impossibile verificare tutti i possibili stati ($10^{189.114}$ per l'HCS08)
- ◆ primo approccio tramite pattern random significativi
- ◆ clock emulato di 0.2MHz (CodeWarrior 0.7MHz, HCS08 40MHz)

Dimostrazione Simbolica

- ◆ cattura tutti i possibili stati
- ◆ impianto dimostrativo rilevante (900 teoremi per l'HCS08)

MATITA COME SUPPORTO PER SPECIFICHE ESEGUIBILI: FORMALIZZAZIONE INTERATTIVA DEI MICROCONTROLLER A 8 BIT FREESCALE



Caratteristiche

- ◆ CPU di tipo CISC
- ◆ supporto al debug
- ◆ controllo di esecuzione e COP
- ◆ 64Kb di RAM/(EP)ROM/FLASH
- ◆ controller di I/O
- ◆ low-end e ultra-low-end

Obiettivi della Specifica

- ◆ 4 CPU (HC05/HC(S)08/RS08)
- ◆ memoria (RAM,ROM,None)
- ◆ 229 modelli con configurazioni specifiche di memoria

MATITA COME SUPPORTO PER SPECIFICHE ESEGUIBILI:
FORMALIZZAZIONE INTERATTIVA DEI MICROCONTROLLER A 8 BIT FREESCALE



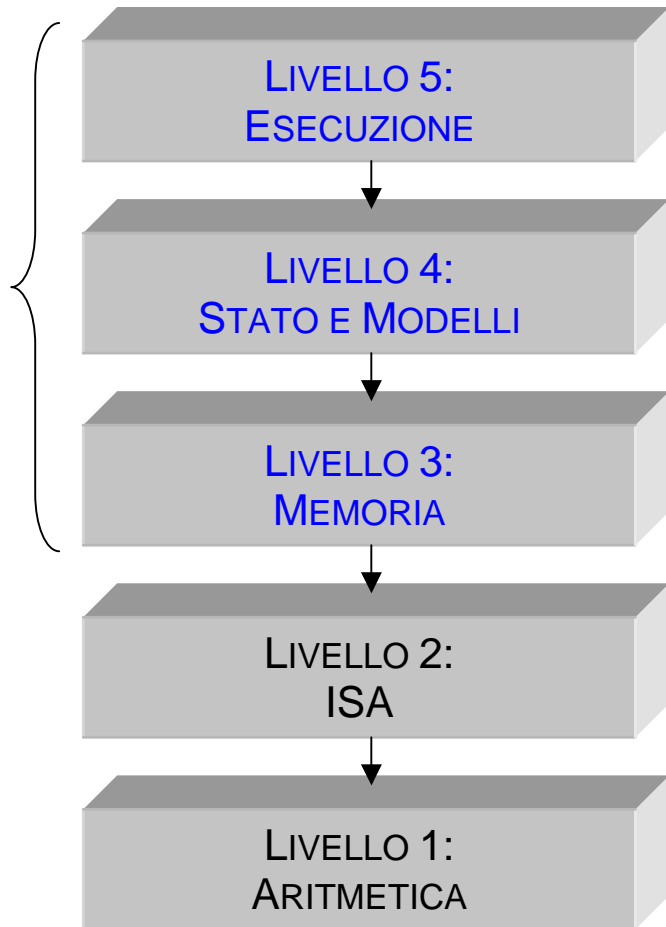
Livello 1 - Aritmetica Modulare [2.728 righe]

- ◆ aritmetica finita in base 16
- ◆ estensione modulare (byte, word)
- ◆ integrazione con l'ISA

Livello 2 - Instruction Set (ISA) [2.497 righe]

- ◆ rappresentazione regolare degli pseudo codici, degli opcode e delle modalità di indirizzamento
- ◆ invarianti di rappresentazione
- ◆ compilazione on-the-fly
- ◆ invarianti di compilazione

MATITA COME SUPPORTO PER SPECIFICHE ESEGUIBILI:
FORMALIZZAZIONE INTERATTIVA DEI MICROCONTROLLER A 8 BIT FREESCALE



Livello 3 - Memoria [1.297 righe]

- ◆ memoria a funzione, ad albero di byte e ad albero di bit
- ◆ paging e memory mapping per l'RS08
- ◆ astrazione di accesso

Livello 4 - Stato e Modelli [1.561 righe]

- ◆ unificazione delle diverse CPU
- ◆ memoria specifica per ogni modello

Livello 5 - Esecuzione [2.162 righe]

- ◆ unificazione delle logiche e delle modalità di indirizzamento
- ◆ esecuzione tick-by-tick, stato di errore e di sospensione

Fase Compilativa [1 mese]

Fase Sperimentale 1 - Specifica Eseguita [2 mesi, 10.300 righe]

- ◆ 25 operatori in base 16, modulari per dimensione (digit, byte e word)
- ◆ 91 pseudocodici e 34 modalità di indirizzamento (4 ad operando implicito)
- ◆ 5 test automatici di consistenza dei dati di esecuzione
- ◆ 229 modelli di microcontroller raggruppati per CPU in 4 famiglie
- ◆ supporto a decodifica e codifica (compilazione)
- ◆ unificazione di 3 diverse implementazioni di memoria
- ◆ unificazione di paging e memory mapping per l'RS08
- ◆ descrizione esatta della RAM, ROM, FLASH equipaggiata da ogni modello
- ◆ unificazione di 48 pseudocodici (su 91) in 10 logiche
- ◆ unificazione di tutte le modalità di indirizzamento
- ◆ supporto all'esecuzione tick-by-tick, allo stato di errore e di sospensione

Fase Sperimentale 2 - Verifica di Frammenti Reali di Codice C [1 mese, 1.900 righe]

- ◆ utilizzo combinato del compilatore CodeWarrior e dell'USB Spider08
- ◆ test in forma di tight/upper bound del tempo di esecuzione e della semantica
- ◆ verifica della corretta esecuzione dei test per un totale di 183 milioni di cicli