

Introduzione al progetto di sistemi hardware con FPGA

Esercitazione 03 di Sistemi dedicati

Docente: Giuseppe Scollo

Università di Catania
Dipartimento di Matematica e Informatica
Corso di Laurea Magistrale in Informatica, AA 2016-17

1 di 10

Indice

1. Introduzione al progetto di sistemi hardware con FPGA
2. argomenti dell'esercitazione
3. struttura di una FPGA
4. flusso di lavoro del progetto con FPGA
5. sintesi automatica su FPGA
6. esperienza di laboratorio
7. uno schematico Quartus II per il datapath del Collatz delay
8. un testbench Quartus II per il datapath del Collatz delay
9. riferimenti

2 di 10

in questa esercitazione si trattano:

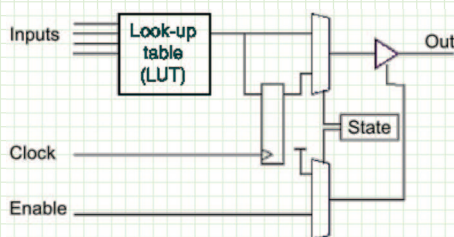
- struttura e organizzazione di una FPGA
- flusso di lavoro tipico di progetto e collaudo su FPGA
- sintesi automatica di circuiti su FPGA
- esperienza di laboratorio

struttura di una FPGA

precursori: PLA, PAL, CPLD

costituenti tipici di una FPGA:

- elementi logici (LE): specializzabili per funzioni logiche definite dall'utente
- blocchi logici complessi (CLB): costituiti da LE, moltiplicatori, registri ecc., configurabili per realizzare componenti circuitali definiti dall'utente
- blocchi per interconnessione, instradamento e I/O



Wilson, Fig. 2.3 - FPGA complex logic block

configurazione del CLB in figura:

assegnamento di una funzione logica alla look-up table (LUT)

funzionamento combinatorio o sincrono

ingresso di abilitazione del pilota di uscita

sequenza di lavoro tipica (non tutti i passi sono presenti in ogni progetto):

1. specifica RTL (progetto di schematico o HDL)
2. analisi sintattica e semantica statica
correzione di eventuali errori, reiterazione dell'analisi
3. simulazione RTL
correzione di eventuali errori semantici, reiterazione di analisi e simulazione
4. sintesi RTL
5. analisi temporale e regolazione del clock
6. sintesi fisica

sintesi automatica su FPGA

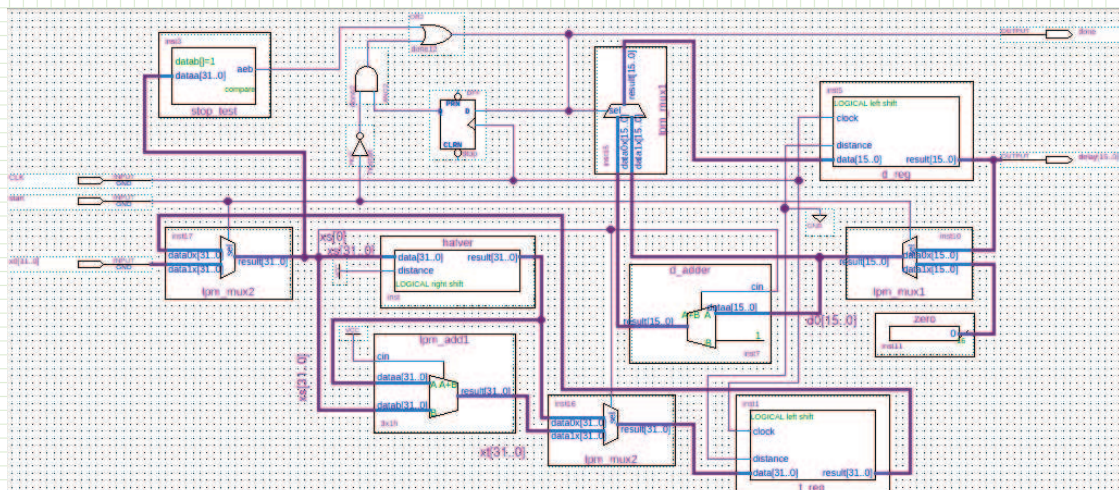
la sintesi fisica, automatizzata da numerosi strumenti di analisi e ottimizzazione, si compone di vari processi:

- traduzione di componenti RTL in componenti FPGA (LE, CLB ecc.)
- dislocazione dei componenti e instradamento dei segnali sulla FPGA
con impiego di opportuni algoritmi di ottimizzazione: recursive cut,
simulated annealing
- generazione della netlist per la FPGA

si possono descrivere, simulare e sintetizzare modelli di circuiti hardware anche senza fare ricorso a un HDL, quando si dispone di un editor grafico per la costruzione dello schematico e di strumenti software adeguati

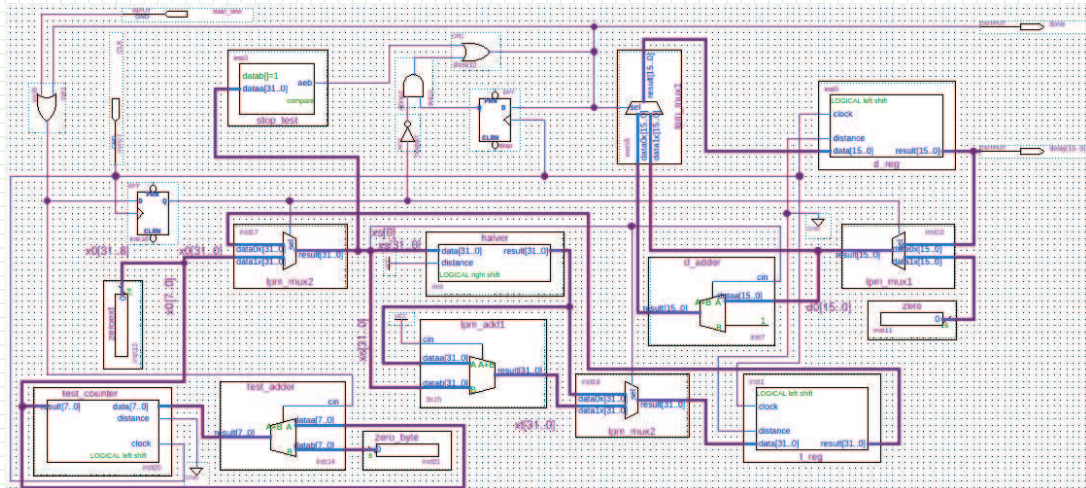
- lanciare Quartus 13.1 (Web edition) e lì creare un nuovo progetto di nome schematic_delay_collatz
- costruire uno schematico del circuito per il delay di traiettorie di Collatz presentato nella seconda lezione ed elaborato nella seconda esperienza di laboratorio
 - un modello che usa diversi moduli parametrici di libreria (lpm) è mostrato nella prossima figura
- compilare e correggere eventuali errori segnalati
- confrontare l'uso di risorse (n. di LE e registri) e slack di caso peggiore con quelli ottenuti dalla compilazione e analisi temporale condotte nella seconda esperienza di laboratorio
- estendere lo schematico con un circuito di collaudo, per pilotare il datapath con i numeri da 1 a 255 quali valori di inizio delle traiettorie
 - l'input per ogni traiettoria successiva alla prima deve attendere che il datapath segnali la fine del calcolo per la traiettoria precedente, v. per esempio la successiva figura
- lanciare l'esecuzione della simulazione funzionale e verificare la corrispondenza del risultato alle attese

uno schematico Quartus II per il datapath del Collatz delay



schematico con uso di moduli parametrici di libreria (lpm)

un testbench Quartus II per il datapath del Collatz delay



un testbench per la simulazione del datapath del Collatz delay

riferimenti

letture raccomandate:

Wilson (2015) Capp. 2, 5, 6

materiali utili per l'esperienza di laboratorio proposta (fonte: Altera University Program, 2014)

Quartus II Introduction Using Schematic Designs - For Quartus II 13.1

Using Library Modules in VHDL Designs - For Quartus II 13.1