

Indice degli appunti

1 Programma del corso **6**

2 Unimap **7**

I Reti combinatorie **11**

3 Venerdì 09/10/2020 **12**
3.1 Introduzione alle reti logiche . . . 12
3.2 Reti combinatorie . . . 15
3.3 Algebra di Boole . . . 21

4 Martedì 13/10/2020 **22**
4.1 Continuiamo con l'Algebra di Boole . . . 22
4.1.1 Proprietà degli operatori di somma e prodotto . . . 23
4.1.2 Porte XOR e XNOR . . . 24
4.1.3 Teoremi di De Morgan con N variabili logiche . . . 25
4.1.3.1 Dimostrazione della prima tesi . . . 25
4.1.3.2 Dimostrazione della seconda tesi . . . 26
4.1.4 Equivalenza tra espressioni dell'Algebra di Boole e reti combinatorie . . . 26
4.1.5 Qualche osservazione sulla risoluzione di esercizi . . . 27
4.1.5.1 Dimostrazione di una identità . . . 27
4.1.5.2 Semplificazione di espressioni . . . 28
4.2 Reti combinatorie significative . . . 30
4.2.1 Decoder . . . 30
4.2.1.1 Decoder 2 to 4 . . . 30
4.2.1.2 Decoder 1 to 2 . . . 31
4.2.1.3 Decoder N to 2^N . . . 32
4.2.2 Decoder con enabler (espandibile) . . . 32
4.2.3 Costruzione di un decoder 4 to 16 da decoder 2 to 4 . . . 33
4.2.4 Demultiplexer . . . 35
4.2.5 Multiplexer . . . 36
4.2.6 Multiplexer come rete combinatoria universale . . . 37

5 Mercoledì 14/10/2020 e Giovedì 15/10/2020 **38**
5.1 Modello strutturale universale . . . 38
5.2 Sintesi di reti in forma SP a costo minimo . . . 41
5.2.1 Metodo algoritmico per la lista degli implicanti principali . . . 42
5.2.2 Metodo di Quine-McCluskey per la lista degli implicanti principali . . . 44
5.2.3 Mappe di Karnaugh . . . 45
5.2.4 Algoritmo di ricerca dei sottocubi principali (implicanti principali) partendo dalle mappe di Karnaugh . . . 47

5.2.5 Ricerca di liste di copertura non ridondanti . . . 48
5.2.5.1 Sintesi di leggi non completamente specificate . . . 50
5.3 Sintesi di porte XOR, XNOR, NAND e NOR in forma SP . . . 52

6 Martedì 20/10/2020 **56**
6.1 Sintesi di reti in formato PS . . . 56
6.1.1 Sintesi duale . . . 58
6.1.2 Sintesi meno costosa . . . 58
6.2 Porte logiche universali . . . 59
6.2.1 Costo a porte e costo a diodi . . . 63

7 Mercoledì 21/10/2020 **64**
7.1 Porte tri-state . . . 64
7.2 Circuiti di ritardo . . . 67
7.2.1 Circuito di ritardo sul fronte di salita . . . 67
7.2.2 Circuito di ritardo sul fronte di discesa . . . 68
7.3 Formatore di impulsi . . . 69
7.3.1 Formatore di impulsi sul fronte di salita . . . 69
7.3.2 Formatore di impulsi sul fronte di discesa . . . 70

II Aritmetica di un calcolatore **71**

8 Mercoledì 21/10/2020 **72**
8.1 Rappresentazione dei numeri naturali . . . 72
8.2 Teorema della divisione con resto . . . 73
8.3 Notazione per indicare quoziente e resto . . . 75
8.4 Proprietà dell'operatore modulo . . . 75
8.4.1 Prima proprietà . . . 75
8.4.2 Seconda proprietà . . . 76
8.4.3 Terza proprietà . . . 76
8.4.4 Quarta proprietà (dal libro di Corsini) . . . 76
8.5 Pippe personali sui segni . . . 77
8.6 Rappresentazione dei numeri in qualunque base β . . . 79
8.7 Rappresentazione di naturali su un numero di cifre finito . . . 79
8.7.1 Quanti numeri possiamo rappresentare? . . . 79
8.7.2 Numero più grande rappresentabile? . . . 80
8.7.3 Numero di cifre richieste per rappresentare un numero . . . 80
8.8 Osservazione sugli esercizi della dispensa . . . 80
8.9 Elaborazione di numeri naturali tramite reti combinatorie . . . 81

9 Giovedì 22/10/2020 **83**
9.1 Complemento . . . 83
9.1.1 Circuito logico per l'operazione . . . 84
9.1.2 Complemento di numeri in base 10 con codifica *eccesso 3* . . . 85
9.2 Moltiplicazione e divisione per una potenza della base . . . 85
9.2.1 Conseguenza: operazioni di concatenamento e scomposizione . . . 86
9.3 Estensione di campo sui numeri naturali . . . 86
9.4 Addizione di numeri naturali . . . 87
9.4.1 Full adder in base 2 . . . 88
9.4.2 Incrementatore (*half adder*) . . . 91
9.5 Sottrazione . . . 93
9.5.1 Osservazione sulla complementazione nei pretest . . . 96

9.5.2 Comparatore di numeri naturali . . . 97
9.6 Moltiplicazione . . . 97

10 Martedì 27/10/2020 **100**
10.1 Moltiplicatore con addizionatore $n \times 1$ in base 2 . . . 100
10.2 Divisione . . . 101
10.2.1 Divisore elementare in base 2 . . . 104

11 Mercoledì 28/10/2020 **105**
11.1 Rappresentazione dei numeri interi . . . 105
11.1.1 Leggi di rappresentazione dei numeri interi . . . 106
11.1.1.1 Modulo e segno . . . 107
11.1.1.2 Traslazione . . . 107
11.1.1.3 Complemento alla radice . . . 108
11.1.2 Proprietà del complemento alla radice . . . 111
11.1.2.1 Determinazione del segno . . . 111
11.1.2.2 Legge inversa . . . 111
11.1.2.3 Forma alternativa della legge . . . 112
11.2 Operazioni su interi in complemento alla radice . . . 113
11.2.1 Valore assoluto . . . 113
11.2.2 Circuito di conversione da CR a MS . . . 114
11.2.3 Calcolo dell'opposto . . . 115
11.2.4 Estensione di campo . . . 117

12 Giovedì 29/10/2020 **120**
12.1 Riduzione di campo . . . 120
12.2 Moltiplicazione per potenza della base . . . 122
12.3 Divisione per potenza della base . . . 122
12.4 Assembler: Shift logico e aritmetico . . . 123
12.5 Somma . . . 124
12.6 Sottrazione . . . 126
12.6.1 Comparazione di numeri interi . . . 126
12.7 Confronto tra comparazioni . . . 127
12.8 Moltiplicazione e divisione . . . 127
12.8.1 Approccio . . . 127
12.8.2 Circuito di conversione da MS a CR . . . 128
12.8.3 Moltiplicazione . . . 129
12.8.4 Divisione . . . 130
12.8.4.1 Condizioni di fattibilità . . . 132
12.8.4.2 Circuito . . . 133

13 Riferimenti ad altri argomenti **134**
13.1 Assembler . . . 134
13.2 Verilog (da *Circuiti logici per le operazioni sui numeri naturali e sui numeri interi* di Paolo Corsini) . . . 136

III Reti sequenziali **142**

14 Martedì 03/11/2020 **143**
14.1 Reti con memoria . . . 143
14.2 Latch SR . . . 144
14.2.1 Pilotaggio di un Lach SR . . . 145
14.3 Il problema dello stato iniziale . . . 146

master copy

14.3.1 Riprendiamo la sintetizzazione del Latch-SR . . . 149
14.4 Tabelle di flusso e grafi di flusso . . . 150
14.4.1 Tabella di flusso . . . 150
14.4.2 Grafi di flusso . . . 151
14.4.3 Diagramma di temporizzazione . . . 151

15 Mercoledì 04/11/2020 **152**
15.1 D-latch trasparente . . . 152
15.2 Reti trasparenti . . . 154
15.3 D flip-flop . . . 154
15.4 Memoria RAM statiche . . . 157
15.5 Montaggio in parallelo di memorie RAM statiche . . . 160
15.6 Montaggio in serie di memorie RAM statiche . . . 160
15.7 Collegamento al bus e maschere . . . 161

16 Giovedì 05/11/2020 **163**
16.1 Memorie ROM (*read-only*) . . . 163
16.1.1 ROM programmabili . . . 165
16.1.1.1 PROM . . . 165
16.1.1.2 EPROM . . . 165
16.1.1.3 EEPROM . . . 166
16.2 Verilog . . . 167
16.3 Reti sequenziali sincronizzate . . . 172

17 Martedì 10/11/2020 **174**

- Descrizione in Verilog di registri.
- Regole di pilotaggio di una RSS, ritardi caratteristici.
- Primo esempio di RSS: Contatore. Descrizione e sintesi in Verilog (con mappe di Karnaugh).
- Contatori e divisione in frequenza.

18 Mercoledì 11/11/2020 **185**

- Registri multifunzionali: descrizione e sintesi in Verilog.
- Esempio di registro multifunzionale.
- Modello di Moore.
- Esempio di rete di Moore: Flip-flop JK. Descrizione e sintesi in Verilog (con mappe di Karnaugh).
- Riconoscitore di sequenze 11, 01, 10. Flip-flop JK come meccanismo di marcatura alternativo al D flip-flop.

19 Giovedì 12/11/2020 **195**

- Modello di Mealy.
- Contatore espandibile in base 3 realizzato come rete di Mealy. Descrizione e sintesi in Verilog.
- Differenze tra reti di Mealy e reti di Moore.



20 Martedì 17/11/2020 **202**

- Modello di Mealy ritardato.
- Osservazioni **vitali** sugli assegnamenti procedurali in Verilog (differenza rispetto al C++).
- Riconoscitore di sequenze come rete di Mealy ritardato.
- RSS complesse: modello generale.
- Primo esempio di RSS complessa: contatore di sequenze corrette 00, 01, 10.
- Linguaggio di trasferimento tra registri: nomenclatura e osservazioni.

21 Mercoledì 18/11/2020 **211**

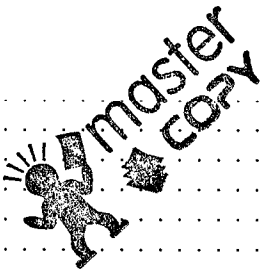
- Esercizio: contatore di sequenze alternate 00,01,10-11,01,10
- Handshake /dav-rfd: regole e formatore di impulsi come esempio.
- Gestione dei cicli in microprogrammazione.

22 Giovedì 19/11/2020 **218**

- Handshake soc-eoc: regole e formatore di impulsi come esempio.
- Sintesi di RSS complesse: metodo della scomposizione in PO/PC.

23 Martedì 24/11/2020 **223**
23.1 Riepilogo sulla sintesi di una RSS compelssa in PO/PC 223
23.2 Tecniche ulteriori per la sintesi della parte controllo 225
23.2.1 Tecniche μ−address e μ−instruction based 225
23.3 Re-introduzione dei μ−salti a più vie: registro MJR 227
23.3.1 Aggiornamento della sintesi PO/PC 228
23.4 Sottoliste utilizzando il registro MJR 229

24 [Stea] Riepilogo sulla descrizione e sintesi di reti logiche **230**



Premesse

Riflessioni Ho riflettuto a lungo sull'opportunità di pubblicare questa dispensa. Chiunque sia interessato ad acquistarla è sicuramente consapevole della difficoltà dell'esame: non esiste la dispensa miracolosa, per superare l'esame di Reti logiche è necessario diventare un tutt'uno con la materia.

Fonti Le fonti di questa dispensa sono:

1. le registrazioni delle lezioni di teoria del prof.Giovanni Stea (con particolare riferimento alle diapositive mostrate durante le lezioni);
2. le dispense rese disponibili dal docente sul suo sito (`http://docenti.ing.unipi.it/~a080368/Teaching/RetiLogiche/materiale.html`, la quasi totalità delle immagini presenti nella dispensa sono prese dal suo materiale);

Perchè pubblicare una dispensa se quelle del docente sono ottime? Ho l'abitudine di scrivere appunti al di la della qualità del materiale fornito. Le dispense sono una rielaborazione sia del materiale fornito dal docente, sia di quanto detto a lezione. **Quindi, perchè dovrei comprare questa dispensa?** Per rendere più veloce la comprensione degli argomenti affrontati. La dispensa non è autosufficiente: dovrà essere utilizzata assieme al materiale fornito dai docenti.

Assembler e struttura del calcolatore? La parte iniziale e finale del corso non sono affrontate in questa dispensa. Sono disponibili due dispense a riguardo su
`http://githubinginfpisa.it/`.