

A

accumulator register, 363
 accuracy, 6
 active-low signals, 153, 253
 address decoder, 249
 address decoding, 257
 address latch, 335
 address lines, 251
 address, memory, 248
 addressing

- immediate, 384
- pointer, 394
- register, 393

 addressing modes, 393
 ADC. *See* analog-to-digital converter
 adders, 141
 AF. *See* auxiliary carry flag
 aliasing, 33
 ALU, 362
 analog, 3, 28
 analog-to-digital converter, 7, 266
 AND gate, 71, 72, 90, 109, 114, 154
 AND rules, 98
 application layer, 305
 arithmetic overflow, 65
 arrays, 385
 assembler, 377
 assembler directive, 380
 assembly language, 340, 341, 346

- comment field, 380
- instruction field, 379
- label field, 378
- operand field, 379

 Associative Law, 96

asynchronous memory, 270
 auxiliary carry flag, 366
 AX. *See* accumulator register

B

base address, 369
 base pointer, 364
 base register, 363
 Basic Input/Output System, 251, 255, 268
 BCD. *See* Binary Coded Decimal
 BCD addition, 62
 binary addition, 41, 141
 Binary Coded Decimal, 38
 binary conversion, 25, 65
 binary pulse, 10
 binary signal, single, 8
 binary signals, multiple, 9
 binary subtraction, 43
 binary system, 8
 BIOS. *See* Basic Input/Output System
 bit, 19, 22
 bitwise AND, 169
 bitwise operations, 168
 bitwise OR, 173
 bitwise XOR, 174
 BIU. *See* bus interface unit
 boolean algebra, laws of, 95
 boolean algebra, simplification, 100
 boolean expressions, 89
 BP. *See* base pointer
 buffer, 331
 bus, 251, 327
 bus contention, 252
 bus interface unit, 367

byte, 22

BX. *See* base register

C

cache

block, 288, 292

direct mapping, 292, 297

fully associative mapping,
292, 297

hit, 291

L1, 287

L2, 287

line, 288

mapping function, 292

miss, 291

set associative mapping,
292, 299

size, 292

split, 287

tag, 288

write back policy, 302

write policy, 292, 301

write through policy, 301

cache replacement algorithm,
292, 297

First In First Out, 298

Least Frequently Used, 298

Least Recently Used, 298

random, 298

capacitor, 269

carry flag, 366

central processing unit, 334

CF. *See* carry flag

checksum, 178

checksum, 1's complement, 179

checksum, 2's complement, 179

chip select, 249, 252, 262

clock, 215, 226

code segment, 371

collisions, 312

combinational logic, 78, 92

Commutative Law, 95

compiler, 377

conditional branching, 329, 390

configuration registers, 335

constant angular velocity, 284

constants, 385

control lines, 251

counter, 218

counter register, 364

CPU. *See* central processing
unit

CRC. *See* cyclic redundancy
check

crosstalk, 307, 310

CS. *See* code segment

CX. *See* counter register

cyclic redundancy check, 182,
311

cylinder, 282

D

D latch, 214, 229, 248, 268

data buffer, 335

data lines, 251

data register, 364

data segment, 371

datagrams, 312

datalink layer, 306, 308, 310

datasum, 178

decode cycle, 348, 372

decoders, 156, 249

DeMorgan's Theorem, 104,
110, 119

demultiplexers, 159

destination index, 365

DF. *See* direction flag

DI. *See* destination index

digital signal processing (DSP),
7
digital value, 4
direct memory access, 358
direction flag, 365
directive. *See* assembler
directive
Distributive Law, 96
divide-by-two circuit, 217
DMA. *See* direct memory
access
don't cares, 137
double word, 22
DRAM. *See* dynamic RAM
DS. *See* data segment
duty cycle, 14
DX. *See* data register
dynamic RAM, 268

E

edge-triggered latch, 216
endian, big/little, 347
embedded system design, 1
ES. *See* extra segment
Ethernet, 310
Ethernet frame
checksum, 311
data, 311
destination address, 311
length, 311
preamble, 310
source address, 311
start delimiter, 311
EU. *See* execution unit
even parity, 177
exclusive-OR gate, 72, 143
execute cycle, 348, 372
execution unit, 362
extra segment, 371

F

falling edge, 10, 207
fetch cycle, 348, 372
flags, 329, 362
flag register, 329
floating-point numbers, 56
formatting, 284
frame, 308
frequency, 13
frequency modulation, 277
full adder, 144

H

half-adder, 142
Hamming Code, 193
header, 308
hertz, 13
hexadecimal, 37
hexadecimal addition, 59
http, 315

I

IC. *See* integrated circuits
ICANN. *See* Internet
Corporation for Assigned
Names and Numbers
IEEE. *See* Institute of Electrical
and Electronics Engineers
IEEE OUI and Company_id
Assignments, 323
IEEE Standard "754", 56
IEEE Standard "802.3", 306,
310, 323
IF. *See* interrupt flag
Institute of Electrical and
Electronics Engineers, 323
instruction pointer, 346, 364
instruction queue, 372
integrated circuits, 161

- Intel assembly
 - ADC, 388
 - ADD, 388
 - AND, 388
 - CALL, 391
 - clearing bits, 392
 - CMP, 391
 - DEC, 388
- Intel assembly (cont...)
 - DIV, 388
 - IN, 387
 - INC, 388
 - INT, 393
 - IRET, 393
 - JMP, 389
 - Jxx, 390
 - LOOP, 391
 - MOV, 387, 388
 - MUL, 388
 - NEG, 388
 - NOP, 393
 - NOT, 388
 - OR, 388
 - OUT, 387
 - PULL, 392
 - PUSH, 392
 - RET, 391
 - SAL, 389
 - SAR, 389
 - setting bits, 392
 - SHL, 389
 - SHR, 389
 - SUB, 388
 - XOR, 388
- Intel directives
 - .CODE, 382, 396
 - .DATA, 382, 396
 - .MODEL, 382
 - .STACK, 382
- DB, DW, DD, DQ, 384
- DUP, 385
- END, 396
- END, 384
- EQU, 385
- PROC, 396
- PROC, 383
- SEGMENT, 380
- Internet Corporation for Assigned Names and Numbers, 323
- internet protocol, 309, 312
- interrupt driven I/O, 356
- interrupt flag, 365
- interrupt service routine, 356
- interrupts, 393, 396
- intersector gap, 281
- intertrack gap, 281
- inverter, 70, 91, 210
- I/O channels, 358
- I/O ports, 335, 373
- I/O processors, 358
- IP. *See* instruction pointer or internet protocol
- IP address, 171, 260
- IP header
 - address fields, 315
 - fragment offset, 314
 - header checksum, 314
 - identification, 313
 - length, 312
 - options, 315
 - padding, 315
 - time to live, 314
 - total length, 313
 - type of service, 313
 - version, 312
- ISR. *See* interrupt service routine

K

Karnaugh map, 126
Karnaugh map rules, 131

L

leakage current, 270
least significant bit, 22, 36, 167
LED. *See* light emitting diode
LIFO, 332
light emitting diode, 14, 148,
164
linker, 377
logic gates, 69
low level formatting, 284
LSB. *See* least significant bit

M

MAC address, 311, 323
machine code, 340
magnetic core memory, 247
maximum, 53
memory
cell, 207
hierarchy, 273
memory (cont...)
internal, 335
map, 255
mapping, 266, 354
space, 256
memory model, 382
minimum, 53
modified frequency
modulation, 277
most significant bit, 22
MP3, 7
MSB. *See* most significant bit
multiple zoned recording, 284
multiplexer, 158

N

NAND gate, 120, 162, 210, 263
NAND-NAND Logic, 119
negative-going pulse, 10
network interface card, 311
network layer, 306, 312, 315
next state truth table, 237
nibble, 22, 36
NIC. *See* network interface card
noise, 7
non-periodic pulse trains, 11
NOT gate. *See* inverter
NOT rule, 96
Nyquist Theorem, 35

O

object file, 377
odd parity, 177
OF. *See* overflow flag
offset address, 369
one's complement, 44
one's complement checksum,
314
one's complement datasum,
179, 321
Open Systems Interconnection
Model, 305
OR gate, 71, 72, 90, 109, 114
OR rules, 97
organization unique identifiers,
323
O/S level formatting, 285
OSI. *See* Open Systems
Interconnection Model
OSI network model, 309
OUI. *See* organization unique
identifiers
output truth table, 237
overflow flag, 366

P

packet, 308
 parallel port, 220
 parity, 177, 193, 197
 parity flag, 366
 partitioning, 285
 pattern detection, 240
 period, 12
 periodic pulse trains, 12
 PF. *See* parity flag
 physical address, 370
 physical layer, 306
 pin "1", 162
 pipelining, 349
 platter, 274
 polling, 355
 positive-going pulse, 10
 powers of 2, multiplication and
 division by, 63
 preamble, 308
 precedence, 92
 prefix, 16
 presentation layer, 305
 principle of locality, 286
 processor status register, 329
 product-of-sums, 114
 program counter, 346
 protoboards, 162
 protocol, 308
 protocol analyzer, 323
 protocol stack, 309
 pull-up resistors, 165
 pulse width, 12
 pulses, 10

R

RAM. *See* random access
 memory
 RAM cache, 287

random access memory, 267
 read enable, 249
 read only memory, 268
 read-write head, 274
 rectangles, 131
 refresh circuitry, 270
 register, 302, 328, 362
 registered ports, 315
 request for comments, 322
 return address, 391
 RFC. *See* request for comments
 rising edge, 10, 207
 ROM. *See* read only memory
 roundoff error, 33
 run length limited, 278

S

S-R latch, 213
 sample, 5
 samples, 7
 sampling rate, 5, 33
 sectors, 281
 segment, 371
 address, 369
 addressing, 368
 registers, 368
 Self-Monitoring Analysis and
 Reporting Technology, 279
 session layer, 305
 seven-segment display, 147
 SF. *See* sign flag
 SI. *See* source index
 sign bit, 48
 sign flag, 366
 signed magnitude, 49, 54
 SMART. *See* Self-Monitoring
 Analysis and Reporting
 Technology
 SOP. *See* sum-of-products

source index, 365
 SP. *See* stack pointer
 spindle, 282
 SRAM. *See* static RAM
 SS. *See* stack segment
 stack, 332
 pointer, 364
 segment, 371
 state, 223
 state diagram, 224
 errors, 228
 state machine, 223, 228
 reset condition, 227, 232
 state transition, 224, 228, 232
 static RAM, 268
 strings, 385
 substrate, 274
 sum-of-products, 109, 125, 129, 155
 switch, 165
 synchronous memory, 271

T

TCP. *See* transmission control protocol
 TCP header
 acknowledgement number, 317
 checksum field, 317
 control bits, 317
 data offset, 317
 destination port, 316
 option field, 318
 sequence number, 316
 source port, 316
 urgent pointer field, 318
 window field, 317
 TCP ports, 315
 TF. *See* trap flag

thrashing, 297
 timing diagram, 77
 track, 280
 trailer, 308, 309
 transistors, 8
 transmission control protocol, 309, 315
 transparent latch, 216
 transport layer, 305, 312, 315
 trap flag, 365
 tristate output, 253
 truth table, 73, 82, 110, 112, 115, 118, 126
 two's complement, 45
 two's complement short cut, 47

U

undefined values, 208
 unsigned binary, 19, 53

V

volatile memory, 251

W

Winchester head, 276, 280
 write enable, 249
 word, 22

X

XOR compare, 175
 XOR gate. *See* exclusive-OR
 XOR subtraction, 185

Z

zero flag, 366
 ZF. *See* zero flag

ABOUT THE AUTHOR

David Tarnoff is an assistant professor in the Computer and Information Sciences Department at East Tennessee State University where he teaches computer hardware, embedded system design, and web technologies. He holds a bachelors and masters of science in electrical engineering from Virginia Tech. In 1999, David started Intermation, Inc., a business that develops software for remote data collection and automation. His research interests include embedded system design and the application of web technologies to teaching and research. David lives in Tennessee with his wife and their son.

NOTE TO THE READER

This textbook was developed after years of teaching computer organization to students of computer science. It incorporates the feedback from hundreds of students and dozens of faculty members and industry professionals. The success of this textbook is a direct result of its users. Therefore, it is important that there always be a direct link between the author and the readers. Please send any feedback you have regarding errors, updates to the material, or suggestions for new material to tarnoff@etsu.edu.

In addition, one of the purposes of this book is to put the concepts of computer organization into the hands of anyone who wants to learn about the topic. As a result, this book should be freely downloadable from the Internet in either its entirety or as individual chapters. If you cannot find a version for download, please e-mail me at tarnoff@etsu.edu, and I will point you to the proper resources.

Thank you for supporting this work.