Errata for Computer Systems Design and Architecture

by Vincent P. Heuring and Harry F. Jordan

Inside front cover, top of left hand page of spread—in the Data Path portion of the figure a double headed arrow should connect the 32 32-bit general purpose registers to the bus (the vertical line in the center of the Data Path section).

Inside front cover: Bottom of right hand page of spread—the labels Condition Code Logic and Shift Control are interchanged. Shift Control should be on the right and Condition Code Logic on the left.

Page 30, problem 1.5 a.—add the words "16 byte" as follows

A certain machine requires 25 μs to process each 16 byte data record in a database.

Page 51, Figure 2.9—delete # symbol from Example column for formats 1 and 7a.

Page 51, Figure 2.9—change shic to shc in format 7.

Page 52, first line of section 2.3.2—change 21 to 23 and 7 to 8. It should read

Figure 2.9 shows 23 instructions in 8 different formats.

Page 52, 8th line from bottom of page and 12th line from bottom of page—change 21 to 23. It should read

12th form bottom: …we define only 23 of these…

8th from bottom: …and then each of the 23 instructions.…

Page 58, Table 2.6—rows 10 and 12 of the table lack entries in the c3⟨2..0⟩ column. They should be 100 and 101, respectively, i.e. copies of the entries immediately above.

Page 58, first line of code below Table 2.6—colon missing after `Cost`, should read

```
Cost:   .equ        125         ;Define symbolic constant
```

Page 61, 5th line from bottom—add a : at end of equation for |x|.

Page 62—the RTN for Instruction Interpretation should read

instruction_interpretation := (
¬Run∧Strt → (Run ← 1; instruction_interpretation):
Run → (IR ← M[PC]: PC ← PC + 4; instruction_execution));

Page 67, Table 2.7, 13th line—reverse 0 and 1 to read

Comparison operators: produce 1 or 0 (true or false) logical value

Page 83—add a sentence to problem 2.4

Assume SRC has a multiply instruction.

Page 85, Problem 2.20 a.—change (op = 29) to (op = 25).

Page 101, last line—w1 should be wl, that is, the symbol after w is lower case L.

Page 103, last sentence in first paragraph—should be “postincrement and predecrement” to read

> The combination of postincrement and predecrement is useful in accessing push-down stacks.

Page 109, 4th line after table—remove : in front of 2nd = to get

move (:= op⟨3..2⟩ = 0) → (

Page 109, 12th line from bottom—change “many” to “may” to read

…a programmer may not wish the flags disturbed…

Page 113, 4th note for Table 3.5—should read

RRR is D register where count is located.

Page 114, 1st note for Table 3.6—should read

`rrr` is one of Dn.

Page 115, Table 3.7—overbars are missing in several of the functions in the Logic column of the table. The corrected logic for the wrong entries is shown below.

Name	**Logic**
CC	$\overline{C}$
HI	$\overline{C}\cdot\overline{Z}$
NE	$\overline{Z}$
PL	$\overline{N}$
VC	$\overline{V}$

Page 123, under RTN for SPARC without Register Windows—the 4th line of the RTN should be modified and a line added under it as follows

r[1..31]⟨31..0⟩:	General registers (current window only);
r[0]⟨31..0⟩ := 0:	Register 0 is always zero.

Page 127, end of paragraph preceding RTN for SPARC Instruction Format—add a sentence.

The use of the asi field is not discussed.

Page 131, second line of RTN for the Subtract Instruction—a space should separate the minus sign form opnd2 at the end of the line

sub (:= op = $10_2 \wedge$ op3 = 000100_2) $\rightarrow$ r[rd] $\leftarrow$ r[rs1] – opnd2:

Page 135, Fig. 3.13, 4th line from end of code—change `%i3` to `%i2` to read

```
        add         %l0, %i2, %l0   ! input registers …
```

Page 135, Fig. 3.13, 2nd line from end of code—change `%o0` to `%i0` to read

```
        restore     %l0, 0, %i0     !Result moved to …
```

Page 139, Problem 3.3—change A0 to PC in the second sentence of the problem

What values would PC, D0, D1, and D2 have after execution?

Page 140, Problem 3.9 b.—change program to subroutine

b. Encode the subroutine in hexadecimal notation.

Page 144, 4th line—add a space between implement and RTN.

Page 145, Figure 4.1—in the Data Path portion of the figure a double headed arrow should connect the 32 32-bit general purpose registers to the bus (the vertical line in the center of the Data Path section).

Page 152, Figure 4.3—remove slant line across Decrement at bottom right of center line.

Page 174, Table 4.15, Step T0 under Control Sequence—replace MA_{in} with MA_{Bin}

PC_{out}, MA_{Bin}, INC4, PC_{in}, Read, Wait

Page 179, 3rd line from bottom of page—change to read

instruction_interpretation := ($\neg$Run$\wedge$Strt $\rightarrow$ (Run $\leftarrow$ 1; instruction_interpretation):

Page 190, reference to M. Morris Mano, *Computer Systems Architecture*, —date should be 1993.

Page 197, last item of bulleted list—change `shr` to `ld`

- The `ld` instruction is being fetched from memory.

Page 202, Figure 5.2—label in the right half of the IR2 box should be c2⟨16..0⟩.

Page 203, Figure 5.3—the second line of the definition of cond should have a ¬ symbol before the second appearance of IR2 and a second) before the : at the end

of the line.

cond := (IR2⟨2..0⟩=1) ∨ ((IR2⟨2..1⟩=1)∧(IR2⟨0⟩⊕R[rc]=0)) ∨
((IR2⟨2..1⟩=2)∧(¬IR2⟨0⟩⊕R[rc]⟨31⟩)):

Page 203, Figure 5.3—the definition of imm needs a second) before the : at the end

imm := addi ∨ andi ∨ ori ∨ (sh ∧ (IR⟨4..0⟩ ≠ 0)) : immediate operand

Page 209, Fig. 5.6, first line in box "2. Decode and operand read"—replace R[rb]

X3 ← l-s2 → (rel2 → PC2: disp2 → ((rb = 0) → 0: (rb ≠ 0) → R[rb]):

Page 210, Figure 5.7—in the equation for multiplexer 1 (Mp1), PC2 should be R1

Mp1 ← (¬(branch2∧cond) → Inc4):
((branch2∧cond) → R1):

Page 210, Figure 5.7—in the equation for Mp4 the string 71mm should be deleted, leaving the last line

(alu∧¬imm → R2):

Page 211, 2nd paragraph, 9th line—c2 = 0 should be c2 ≠ 0 and in last line of paragraph, c2 ≠ 0 should be c2 = 0.

Page 226, Eq. 5.1—subscript in denominator of first right hand side should be w instead of w/o

$$\% \text{ Speedup } = \frac{(IC \times CPI \times \tau)_{w/o} - (IC \times CPI \times \tau)_{w}}{(IC \times CPI \times \tau)_{w}} \times 100 = \frac{CPI_{w/o} - CPI_{w}}{CPI_{w}}$$

Page 234, Figure 5.18—the second AND gate from the top on the left side of the figure should have an inversion bubble on its top input, as does the 4th AND gate.

Page 236, Table 5.3, row for Address 204, under Branching Action—change N to $\overline{N}$

`To 206 if` $\overline{N}$ `(else 205)`

Page 238, Figure 5.20—in the decoder, change the n in 2n to a superscript to get

n to 2^n decoder

Page 247, Problem 8—second line of code has digits 0 and 3 reversed. it should be

```
404          brge  r3, r0
```

Page 247, Problem 9 b. 6.—replace `31` by `r31` to read

```
br r31
```

Page 249, first line—insert the word operations

b. Why is he sure that all three operations can be grouped together in F7?

Page 295, Table 6.7—last line of table in Value column starts with 1. instead of 0. It should read

$$(-1)^s \times (0.f_1\, f_2 \ldots) \times 2^{-126}$$

Page 373, reference to H. G. Cragon— “Bartlerr” should be Bartlett.

Page 406, Example 8.3—heading should read

Extract the Correct Data Value from the Odd-Parity SECDED String 0110101101101

Page 410, Problem 8.10—should have section reference (§8.3)

Page 430, Figure 9.11—a horizontal line should start at the top of the arrow going into the top of the Shift register box, extend through the top of the arrow going into the top of the top Incr. box and extend to the right with a label Dot clock

Page 438, Figure 9.17—remove arrow extending to the right from the dot above the label Analog comparator and remove the dot. Add a right pointing arrow from the middle of the right side of the Successive approximation logic box and move the label Done from the deleted arrow to this new one.

Page 440, Figure 9.19—in the Quantization error part of the figure, both appearances of 8 should be 16.

$$+\frac{Vf}{16} \text{ and } -\frac{Vf}{16}$$

Page 474, Figure 10.14, part (a)—change “256 subnets” to 254 subnets.

Page 524, Figure A.61—the labels Q and $\overline{Q}$ at the outputs of the Circuit are reversed and the connections feeding back from these points to the leftmost pair of AND gates should connect to the outputs of the opposite NORs.

Page 526, Figure A.63—the labels q_0 and q_1 at the outputs of the 2-bit synchronous counter box are reversed.

Page 546, second line of RTN under Instruction Interpretation—should read

$\neg$Run$\wedge$Strt $\rightarrow$ (Run $\leftarrow$ 1; instruction_interpretation):

Page 546, 4th line under Instruction Execution—remove (rb) to get

str (:= op= 4) $\rightarrow$ M[rel] $\leftarrow$ R[ra]: str ra, offset Store reg. rel.

Page 549, last line of RTN under Instruction Interpretation—add one more) before the : to get

PC ← PC + 4; instruction_execution)):

Inside back cover, left side of spread, under Category—dividing line between Shift and Branch is one row too high. It should separate

`shc ra, rb, rc`

from

`br rb, rc, c3`